Advances in Particle/Finite Volume Algorithms for Turbulent Reactive Flows

A Dissertation
Presented to the Faculty of the Graduate School
of Cornell University
in Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

Pavel Petkov Popov

January 2013

Advances in Particle/Finite Volume Algorithms for Turbulent Reactive Flows

Pavel Petkov Popov, Ph.D.

Cornell University 2013

In the field of turbulent reactive flow simulations, hybrid particle/finite volume large eddy simulation/probability density function (LES/PDF) methods have been shown to be highly accurate in simulating laboratory-scale flames. Their strengths lie in the combination of the large eddy simulation procedure's ability to resolve the large, non-universal scales of turbulence, combined with the fact that probability density function models for turbulent combustion require no closure for the highly non-linear chemistry source term.

This work presents advances in such hybrid particle/finite volume LES/PDF algorithms for turbulent reactive flows. New time stepping, interpolation, and coupling schemes have been proposed with the goal of reducing particle mass consistency (PMC) error (defined as the discrepancy between particle mass density and resolved finite volume density) and overall simulation error.

The Multi-step Second-order Runge-Kutta (MRK2) integration scheme is an ODE integration scheme designed for reducing PMC errors when applied to discontinuous velocity fields. When applied to a discontinuous velocity field such as might be produced by a state-of-the art velocity interpolation scheme, MRK2 preserves the continuity of the Lagrangian position mapping and is second-order convergent in time, as opposed to a standard second-order Runge-Kutta scheme, which is only first-order convergent in time when applied to a discontinuous velocity field.

The Direct Richardson p-th order (DRp) is a conceptually new family of SDE integration schemes which are weakly p-th order accurate in time, where p is an arbitrary positive integer. Unlike standard SDE integration schemes, which are based on matching appropriate terms in the Ito-Taylor expansion of the stochastic process, the DRp schemes work via Richardson extrapolation between the probability density functions of a set of first-order accurate Euler approximations with differing time steps.

In the context of the Large Eddy Simulation/Probability Density Function (LES/PDF) code developed by the Turbulence and Combustion Group at Cornell University, a PDF to LES density coupling scheme via a transported specific volume (TSV) has been developed. While coupling approaches similar to TSV have been used previously in LES/PDF application, the present implementation is the first to allow overall second-order accuracy of the LES/PDF code in space and time.

New implicit and explicit schemes for PMC error reduction schemes have been developed and tested in the context of the Sandia-Sydney bluff-body flame. Implicit PMC preservation schemes include new velocity and diffusivity interpolation algorithms, and explicit PMC error correction is achieved via a corrective velocity. While corrective velocity schemes have been used previously, the present algorithm, featuring a smoothed version of the PMC error field, is

capable of maintaining the same PMC error levels with a corrective velocity of lower magnitude.

Finally, the LES/PDF algorithm, developed by the Turbulence and Combustion group at Cornell, is applied to the Sandia-Sydney bluff-body flames. Comparison is made with experimental data, and the new code is in better agreement with experiment than previous simulations of the same series of flames.

BIOGRAPHICAL SKETCH

Pavel Popov was born on April 22, 1984 in Sofia, Bulgaria. He came to the United States in August 2003 to study Mechanical Engineering at Cornell University. Receiving his B.S. in Mechanical Engineering in January 2007, he enrolled in the Aerospace Engineering Ph.D. program under the tutelage of Prof. Stephen B. Pope. While in the Ph.D. program, he received an M.S. (Special) in December 2009. His dissertation deals with transport and specific volume coupling in particle/finite volume algorithms for turbulent reactive flows.

To my family and to my teachers

ACKNOWLEDGMENTS

I would like to thank Prof. Stephen Pope for his guidance during my Ph.D. research, his patience and his way of leading by example. I also thank Prof. David Caughey for the many insights that he shared with me during my time here. Many thanks to Prof. Alexander Vladimirsky, who guided me through my minor in Mathematics.

I would not have gotten involved in the field of turbulent reactive flows were it not for Profs. John Hubbard and Alexander Valdimirsky, who sparked my interest in this field by mentioning just how challenging the Navier-Stokes equations are; this interest was further strengthened by (in chronological order) Profs. Charles Williamson, Zellman Warhaft, Stephen Pope, and Kenneth Torrance, who taught me the theory of the Navier-Stokes equations and methods for their solution, in both incompressible and reactive settings - my thanks to all of the above.

I am also very grateful to Prof. Michel Louge and the dearly departed Prof. Kenneth Torrance, who guided me during my teaching experiences in the Ph.D. program.

I am grateful for the support of all the members of the Turbulence and Combustion Group at Cornell, particularly Haifeng Wang, Sharadha Viswanathan and Steve Lantz.

Last but not least, many thanks to all the brothers at the Seal and Serpent society, and all my friends, both two- and four-legged, at the Oxley equestrian center.

TABLE OF CONTENTS

1 Introduction

Ubiquitous in engineering applications, from internal combustion engines, through power plant and jet gas turbines, rocket engines, and even environmental phenomena such as forest fires, turbulent combustion processes are an essential component of modern engineering science. As with all physical processes, the ability to predict their behavior is not only the ultimate form of their understanding, but also highly advantageous in the future design of applications which incorporate turbulent combustion - better predictive capability at the design stage will allow for a more efficient, stable and clean final product.

Because of this, computational methods for turbulent combustion have been a vibrant area of research for over four decades [19,20,3,27]. The challenge in performing accurate computations of turbulent reactive flows lies in the considerable separation between the largest and smallest scales of motion, which necessitates the use of extremely refined grids if an exact solution to the Navier-Stokes and chemical evolution equations is to be achieved. Therefore, a model for the unresolved turbulent structures is required for most flows to be computationally tractable [21].

Building on the well-developed theory of incompressible turbulent flow [8, 9, 24, 10, 4, 6, 21], and on established mechanisms of chemical kinetics for predicting the evolution of chemical compositions [25,26], computational methods for turbulent combustion fall into many different classes depending on the type of modeling used for the turbulent motions and turbulence-chemistry interactions. Given sufficient computational resources, it is also possible to perform a direct numerical simulation (DNS) of turbulent combustion, but even with state-of-the art technology, the high cost of DNS limits this option to a small fraction of turbulent reactive flows with low to moderate Reynolds numbers [2].

Provided that some turbulence modeling is used, turbulent combustion simulations are either of the Reynolds-averaged Navier-Stokes (RANS) [6], or of the Large Eddy Simulation (LES) [24] type. The major difference between the RANS and LES types of simulations is that all turbulent motions are modeled in RANS, whereas in LES only the smallest (and, according to the Kolmogorov hypotheses, universal) scales are modeled.

An additional distinction between different methods for turbulent reactive flow is due to the different models for turbulence-chemistry interactions (TCI), especially the approach used for modeling the unresolved chemical source term, which is challenging due to the source term being highly non-linear as a function of temperature [18,16]. The simplest of these models, applicable in a very limited number of cases and completely inadequate in most situations [18] is the so-called Arrhenius, or "no-model" model, which assumes that the mean of the source term equals the source term corresponding to the composition mean. Other TCI models such as the flamelet-assumed PDF [17] yield better results over a broader range of cases, but still require that assumptions be made for the unresolved source term.

In the field of turbulent reactive flow calculations, the Probability Density Function (PDF) method for TCI modeling [19] has an advantage over most other

TCI models in that it requires no closure for the highly non-linear chemistry source term. First used in conjunction with Reynolds-averaged Navier-Stokes (RANS) hydrodynamic solvers, the resulting RANS/PDF class of algorithms are to this date effectively used in the simulation of turbulent reactive flows [11,14].

In the past two decades, with increases in computational resources, Large Eddy Simulation (LES) methods have been increasingly supplanting RANS as the hydrodynamic solvers to be used in conjunction with PDF chemistry solvers. The resulting class of LES/PDF algorithms, first introduced by Pope [20], have the usual benefits over RANS solutions in that they do not require modeling of the large scales of turbulent motion, and can capture time-dependent flow phenomena, such as vortex shedding, for example. Hybrid LES/PDF methods have been demonstrated to be highly effective in simulating laboratory scale flames [3,5,23,22].

In a standard LES/PDF application, the LES code is a finite volume solver which evolves the momentum and continuity equations, with an appropriate turbulence model for the unresolved motions. An example of such a code is the one developed by Pierce and Moin [17], a modification of which is used in a large portion of the present work. The other component of an LES/PDF algorithm is a Lagrangian particle PDF code, which provides a Monte Carlo solution to the evolution equation of the joint PDF of chemical compositions. This Monte Carlo solution is evaluated from an ensemble of Lagrangian particles, which are advected in physical space by a stochastic differential equation (SDE), whose diffusion term is determined by the LES turbulent diffusivity and whose drift term is determined by both the LES resolved velocity and turbulent diffusivity.

The hybrid nature of LES/PDF methods poses several challenges. One of these is that several fields are defined in two separate forms - as LES fields and as properties of the particle ensemble; as pointed out by Muradoglu et al. [15], correspondence between these redundantly defined fields is necessary for the LES/PDF procedure to be valid. Another issue arising from the disparate nature of the two components of an LES/PDF code is that certain fields communicated from the PDF to the LES component contain statistical error due to the stochastic nature of the former; in the case of the density field, whose time derivative is needed by the LES code in the solution of the continuity equation, this statistical error may be the cause of significant errors and even instability in the overall solution.

In this dissertation, we focus on two aspects of the LES/PDF procedure which are closely connected to the issues described in the above paragraph. The first of these is the evolution of particles in physical space, with a special emphasis on the preservation of consistency between the mass of particles in a given region (which is the form of the density field carried by the particle ensemble), and the LES finite-volume density field - we shall henceforth refer to this as particle mass consistency (PMC). Secondly, we consider the transfer of specific volume information from the PDF to the LES code, with the goal of minimizing the errors in the overall solution caused by the stochastic nature of the PDF code.

The main body of the dissertation is divided into five chapters, each of which consists of a paper already published, in review, or in preparation for submission to a scientific journal.

Chapter 2: *An Accurate Time Advancement Algorithm for Particle Tracking* (published in the Journal of Computational Physics) presents a new time integration scheme for ordinary differential equations (ODE) which is designed to deal with spatial discontinuities in the velocity field by which the particles are advected. It builds on the work of McDermott and Pope [13], who introduce a new velocity interpolation scheme, called the Parabolic Edge Reconstruction Method (PERM) specifically designed to preserve PMC by extending the second-order multilinear scheme for velocity interpolation so that the divergence of the interpolated velocity has certain desirable qualities. While the PERM scheme has been shown in [13] to improve particle mass consistency, it also features discontinuities along the faces of the finite volume cells, which can be problematic when a standard ODE integration scheme is applied to the resulting velocity field. In particular, when the second-order Runge-Kutta (RK2) ODE integration scheme is applied to a PERM velocity field on a fixed grid, the accuracy of the scheme with respect to the time step decreases to first-order, and the continuity of the Lagrangian mapping between initial and final particle positions is violated. The new time integration scheme introduced in Chapter 2, called the modified second-order Runge-Kutta (RK2), rectifies these issues: it is second-order accurate in time, preserves the continuity of the Lagrangian mapping, and is shown to better satisfy PMC when applied to a known analytic velocity field.

Chapter 3: *The Direct Richardson p-th Order (DRp) Schemes - A New Class of Time Integration Schemes for Stochastic Differential Equations* (published in the SIAM Journal on Scientific Computing), introduces a new class of weak $p-$th order (for an arbitrary positive integer p) schemes for the integration of stochastic differential equations. Unlike standard SDE integration schemes, which rely on matching terms up to a given degree in the Ito-Taylor expansion of a stochastic process, the DRp schemes perform an acceptance-rejection based Richardson extrapolation of the PDF of the final particle position after at the end of a time step, as given by first-order accurate Euler solutions with smaller time steps. The DRp SDE integration schemes are tested on a 2D anisotropic and a 1D isotropic test case, and their performance is compared to that of the schemes of Kloeden and Platen [7] and Cao and Pope [1]. The error yielded by the second- and third-order members of the DRp class of schemes is lower than that of [7,1], whereas the computational cost is slightly higher, so that the overall computational efficiency is comparable. Thus, the DRp schemes are a practicable alternative to other SDE integration schemes, with the benefit of being easier to imlpement.

Chapter 4: *Specific Volume Coupling and Convergence Properties in Hybrid Particle/Finite Volume Algorithms for Turbulent Reactive Flows* (in review by the Journal of Computational Physics) deals, in the context of the LES/PDF code developed by the Turbulence and Combustion Group at Cornell (called LES/HPDF), with the transfer of specific volume information from the PDF to

the LES component of the code, and the implementation of LES to PDF coupling which is overall second-order accurate in time (previous LES/PDF codes are only first-order accurate in time, although they may feature higher-order spatial discretization). Two alternative schemes for specific volume coupling are tested, and their performance is evaluated based on the amount of coupling error introduced in an LES/PDF simulation of the Sandia-Sydney Bluff-Body flame HM1 [12]. A new LES/PDF time-stepping procedure, designed for second-order accuracy with respect to the time step, is introduced, and the overall second-order convergence of the LES/PDF code, with respect to both grid spacing and time step, is verified on a well-resolved vortex ring flow.

Chapter 5: *Implicit and Explicit Schemes for Mass Consistency Preservation in Hybrid Particle/Finite Volume Algorithms for Turbulent Reactive Flows* (in review by the Journal of Computational Physics) introduces new algorithms for the preservation of PMC, and tests them in the context of the LES/HPDF code. A new velocity interpolation scheme called PPERM, which is an extension of the PERM interpolation scheme to cylindrical coordinates, is introduced, as well as a new, third-order accurate scalar interpolation scheme, called the Multilinear Gradients scheme (MLG). In the context of the Sandia-Sydney Bluff-Body flame HM1, it is shown that the use of PPERM/MLG for the interpolation of LES fields gives an improvement of PMC when compared the standard multilinear and fourth-order Lagrangian interpolation schemes.

Numerical tests also demonstrate a reduction in the PMC error when a second-order accurate SDE integration scheme is used for particle tracking, rather than the first order Euler scheme; while that is intuitive, it is also the first time that a second-order accurate time stepping scheme has been demonstrated to hold advantages over a first-order scheme in a practical LES/PDF application. Additionally, alternative schemes for the explicit preservation (via an additional corrective velocity) of particle mass consistency are tested, with the optimal among them meeting the desired PMC criterion with a low level (around 5% relative magnitude) of the corrective velocity.

Finally, Chapter 6: *Large Eddy Simulation/Probability Density Function Solutions of a CH4/H2 Bluff Body Stabilized Flame* (in review by "Combustion and Flame") presents solutions of the Sandia-Sydney Bluff-Body flames, using the latest version of the LES/HPDF code with chemistry described by the 19-species ARM2 chemical mechanism. Unlike the previous two chapters, where the bluff-body flame was used simply as a representative LES/PDF context, in this chapter we make a comparison between the LES/HPDF solution, experimental measurements and previous calculations of the same flow, using both the RANS/PDF and LES/PDF approaches. The LES/HPDF results compare favorably with other simulations: while the results indicate that the outer shear layer is somewhat under-resolved, in the downstream regions of the flow the agreement with experiment is better than that of previous computational studies for the bluff-body flames.

References

[1] R. Cao, and S. B. Pope, *Numerical Integration of Stochastic Differential Equations: Weak Second-Order Mid-Point Scheme for Application in the Composition PDF Method*, Journal of Computational Physics **185**, 194-212 (2003)

[2] J.H. Chen, *Petascale Direct Numerical Simulation of Turbulent Combustion-Fundamental Insights Towards Predictive Models*, Proc. Comb. Inst. **33** (2011), pp. 99-123

[3] P. Colucci, F. Jaberi, P. Givi, and S. B. Pope, *Filtered Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, Phys. Fluids **10** (1998), pp.499-515

[4] M. Germano, U. Piomelli, P. Moin, and W.H. Cabot *A dynamic subgrid-scale eddy viscosity model*, Phys. Fluids, **3**, pp. 1760-1765, 1991

[5] F. Jaberi, P. Colucci, S. James, P. Givi, and S. B. Pope, *Filtered Mass Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, J. Fluid Mech. **401** (1999), pp.85-121

[6] W.P. Jones, and B.E. Launder, *The Prediction of Laminarization with a Two-Equation Model of Turbulence*, Int. J. Heat Mass Transfer **15** (1972), pp. 301-314

[7] P. E. Kloeden, and E. Platen *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, Berlin, 1992

[8] A.N. Kolmogorov, *Dissipation of Energy in Locally Isotropic Turbulence*, Dokl. Akad. Nauk SSSR **32** (1941), pp.19-21

[9] A.N. Kolmogorov, *Dissipation of Energy in Locally Isotropic Turbulence*, Dokl. Akad. Nauk SSSR **30** (1941), pp.299-303

[10] D.K. Lilly, *The Representation of Small-Scale Turbulence in Numerical Simulation Experiments*, Proc. IBM Sci. Comp. Symp. on Env. (1967)

[11] K. Liu, S. B. Pope, and D. A. Caughey, *Calculations of Bluff-Body Stabilized Flames Using a Joint Probability Density Function Model with Detailed Chemistry*, Comb. Flame, **141**, (2005), pp. 89-117

[12] A. R. Masri, and R. W. Bilger, *Turbulent Diffusion Flames of Hydrocarbon Fuels Stabilized on a Bluff Body*, Proc. Comb. Inst., **20**, (1985)

[13] R. McDermott, and S.B. Pope, *The Parabolic Edge Reconstruction Method (PERM) for Lagrangian Particle Advection*, J. Comput. Phys., **227** (2008), pp. 5447-5491

[14] B. Merci, D. Roekaerts, B. Naud, and S. B. Pope, *Comparative Study of Micromixing Models in Transported Scalar PDF Simulations of Turbulent Nonpremixed Bluff Body Flames*, Comb. Flame, **146**, (2006), pp. 109-130

[15] M. Muradoglu, S. B. Pope, and D. A. Caughey, *The Hybrid Method for the PDF Equations of Turbulent Reactive Flows: Consistency Conditions and Correction Algorithms*, J. Comput. Phys., **172** (2001), pp. 841-878

[16] N. Peters, *Turbulent Combustion*, Cambridge University Press (2000)

[17] C. D. Pierce, and P. Moin, *Progress-Variable Approach for Large-Eddy Simulation of Turbulent Combustion*, J. Fluid Mech.**504**, (2004), pp. 73-97

[18] T. Pointsot, and D. Veynante, *Theoretical and Numerical Combustion* 2nd ed., Edwards (2005)

[19] S. B. Pope, *PDF Methods for Turbulent Reactive Flows*, Prog. Energy Combust. Sci., **11**, pp. 119-192 (1985)

[20] S. B. Pope *Computations of Turbulent Combustion: Progress and Challenges*, Proc. Comb. Inst., **23**, (1991), pp. 591-612

[21] S. B. Pope, *Turbulent Flows*, Cambridge University Press, Cambridge, 2000

[22] V. Raman, H. Pitsch, and R. O. Fox, *Hybrid Large-Eddy Simulation/Lagrangian Filtered-Density-Function Approach for Simulating Turbulent Combustion*, Comb. Flame, **143**, (2005), pp. 56-78

[23] V. Raman, and H. Pitsch, *A Consistent LES/Filtered Density Function Formulation for the Simulation of Turbulent Flames with Detailed Chemistry*, Proc. Comb. Inst., **31**, (2007), pp. 1711-1719

[24] J. Smagorinsky, *General Circulation Experiments with the Primitive Equations: I. The Basic Equations*, Mon. Weather. Rev. **91** (1963), pp. 99-164

[25] G.P. Smith, D.M. Golden, M. Frenklach, N.W. Moriarty, B. Eiteneer, M. Goldenberg, C.T. Bowman, R.K. Hanson, S. Song, W.C. Gardiner, Jr., V.V. Lissianski, and Z. Qin *http* : *//www.me.berkeley.edu/gri_mech/*

[26] Q. Tang, Ph.D. Thesis, Cornell University, 2003

[27] J. Villermaux, and J.C. Devillon *Reprèsentation de la redistribution des domaines de sègrègation dans un fluide par un modèle d'interaction phènomènologique*, 2nd Int. Symp. Chem. React. Engng, Amsterdam, 1972

An Accurate Time Advancement Algorithm for Particle Tracking

Pavel P. Popov
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

Randall McDermott
Building and Fire Research Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8663, USA

Stephen B. Pope
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

January 2008

Abstract

We describe a particle-position time-advancement algorithm that is designed for use with several subgrid velocity reconstruction schemes used in LES/FDF methods, and potentially in other applications. These reconstruction schemes yield a subgrid velocity field with desirable divergence properties, but also with discontinuities across cell faces. Therefore, a conventional time advancement algorithm, such as second-order Runge-Kutta (RK2), does not perform as well as it does with a smooth velocity field. The algorithm that we describe, called Multi-Step RK2 (MRK2), builds upon RK2 by breaking up the time step into two or more substeps whenever a particle crosses one or more velocity discontinuities. When used in conjunction with the Parabolic Edge Reconstruction Method, MRK2 performs considerably better than RK2: both the final position of an advected particle, and the final area of a 2D infinitesimal area element are second-order accurate in time (as opposed to first-order accurate for RK2). Furthermore, MRK2 has the theoretical advantage that it better preserves the continuity of the mapping between initial and final particle positions.

1 Introduction

In this paper, we consider some aspects of particle tracking in hybrid finite-volume/particle PDF methods for turbulent reactive flows. In these methods, and potentially in other CFD applications, we have a large number of particles (on the order of 10^7) whose positions are initially random and evolve by the ODE

$$\frac{\mathrm{d}\mathbf{X}^*}{\mathrm{d}t} = \mathbf{U}^* = \mathbf{U}\left(\mathbf{X}^*(t), t\right), \tag{1}$$

where $\mathbf{X}^*(t)$ is a particle's position as a function in time, and $\mathbf{U}\left(\mathbf{X}^*(t), t\right)$ is the velocity experienced by the particle. Usually, this velocity consists of a deterministic component and a random term which is part of the turbulence model: in the present paper, we consider the deterministic part only. In general, velocity information is available at discrete locations on a finite-volume (FV) grid, and therefore a particle tracking algorithm consists of two parts: a velocity interpolation scheme which interpolates the FV velocity onto the particle locations, and a time advancement scheme which updates the particle locations, using the interpolated velocity.

In recent research on FV/particle PDF methods for reacting flows, accurate particle tracking has been recognized as an important condition for maintaining numerical consistency between the finite–volume and the particle aspects of the solution. Muradoglu et al. [2] define mean particle mass density, $q(\mathbf{x}, t)$, as the expectation of the total mass of particles in an infinitesimal region, normalized by that region's volume:

$$q \equiv \langle m^* \delta(\mathbf{X}^* - \mathbf{x}) \rangle \tag{2}$$

(where m^* denotes a particle's mass), and have shown that an important consistency condition is that the mean particle mass density should remain equal to the FV mean density:

$$q = \langle \rho \rangle, \tag{3}$$

provided that the initial conditions are consistent, $q\left(\mathbf{x}, 0\right) = \langle \rho\left(\mathbf{x}, 0\right) \rangle$. The meaning of Eq.(3) is that the expected mass of all particles inside a certain region S must be equal to the mass of S as given by the FV density field. In the incompressible case, for example, this means that if particles are initially uniformly distributed in the computational domain, then they should remain so for all time.

In order to satisfy the above consistency condition, Muradoglu et al. [2] employ a position-correction algorithm which introduces a small displacement in the particle positions after each time step, in order to enforce Eq.(3). A similar position-correction algorithm has been employed by Zhang and Haworth [4].

In a different approach, Jenny et al. [3] note that if particles move with a velocity

$$\mathbf{U}^* = \tilde{\mathbf{U}} + \mathbf{u}^*, \tag{4}$$

where $\tilde{\mathbf{U}}$ is a deterministic component interpolated from the FV velocity, and $\mathbf{u}^*$ is a random component with zero mean, then the following equation holds:

$$\left(\frac{\partial}{\partial t} + \tilde{\mathbf{U}} \cdot \nabla\right) \ln q = -\nabla \cdot \tilde{\mathbf{U}}, \tag{5}$$

which has the same form as the mean continuity equation

$$\left(\frac{\partial}{\partial t} + \langle \mathbf{U} \rangle \cdot \nabla\right) \ln \langle \rho \rangle = -\nabla \cdot \langle \mathbf{U} \rangle , \tag{6}$$

where $\langle \mathbf{U} \rangle$ is the mean velocity. Therefore, if Eq.(3) is satisfied at $t = 0$, it will also be satisfied implicitly at future time, provided that the velocity interpolation scheme yields accurate values for the reconstructed velocity and its divergence, and provided that an accurate time advancement scheme is used. Addressing the velocity interpolation issue, Jenny et al. [3] introduce a 2D velocity interpolation scheme with desirable divergence properties. McDermott and Pope [1] improve upon this scheme, and extend it to 3D, calling the new scheme the Parabolic Edge Reconstruction Method (PERM). It has been shown [1] that PERM performs better than standard multilinear interpolation in satisfying the above-mentioned consistency condition in the particle tracking limit (i.e., when there are no velocity fluctuations).

In the present work, we make a further improvement in particle tracking by using a time-stepping algorithm which has been specifically designed for use in conjunction with PERM. The new algorithm, called Multi-Step Runge-Kutta 2 (MRK2) is quite similar to the standard second-order Runge-Kutta (RK2), but it provides a more accurate treatment of particles which cross a velocity discontinuity. Although MRK2 is motivated by the PERM reconstruction, it can also be used as an alternative to RK2 in all applications in which particles need to be advected though a velocity field with discontinuities at known locations.

To illustrate the benefits of PERM and MRK2, Figure 1 shows final particle distributions for a simple 2D incompressible test flow with a uniform initial particle distribution. In this case, the final particle distribution should be uniform as well. It can be seen on Figure 1 that, when RK2 is used as the time advancement scheme, the PERM velocity interpolation yields a more uniform final particle distribution than standard bilinear interpolation. We can also see that the final particle distribution becomes even more uniform when the new time-stepping scheme, MRK2, is used in conjunction with the PERM velocity interpolation.

Properties of the PERM velocity reconstruction

The reader is encouraged to refer to [1] for a thorough decription of PERM and its properties–here we focus only on its aspects which are relevant to the MRK2 time-stepping scheme.

The PERM velocity reconstruction scheme uses discretized velocity information in the form of face-average velocity components in the direction normal to a given grid cell face (we shall refer to this as the "FV velocity", for the sake of brevity). The PERM reconstructed velocity within a given grid cell depends only on the FV velocity on the faces of that cell and its nearest neighboring cells. Its functional form, in two dimensions (the extension to 3D is rather intuitive) is:

$$\begin{aligned}
u(q,r) &= (1-r)\left[\bar{u}_S + \left(q-\tfrac{1}{2}\right)\Delta^{(1)}_{u,S} + \tfrac{1}{2}\left\{\left(q-\tfrac{1}{2}\right)^2 - \tfrac{1}{4}\right\}\Delta^{(2)}_{u,S}\right] \\
&+ \quad r\left[\bar{u}_N + \left(q-\tfrac{1}{2}\right)\Delta^{(1)}_{u,N} + \tfrac{1}{2}\left\{\left(q-\tfrac{1}{2}\right)^2 - \tfrac{1}{4}\right\}\Delta^{(2)}_{u,N}\right] \\
v(q,r) &= (1-q)\left[\bar{v}_W + \left(r-\tfrac{1}{2}\right)\Delta^{(1)}_{v,W} + \tfrac{1}{2}\left\{\left(r-\tfrac{1}{2}\right)^2 - \tfrac{1}{4}\right\}\Delta^{(2)}_{v,W}\right] \\
&+ \quad q\left[\bar{v}_E + \left(r-\tfrac{1}{2}\right)\Delta^{(1)}_{v,E} + \tfrac{1}{2}\left\{\left(r-\tfrac{1}{2}\right)^2 - \tfrac{1}{4}\right\}\Delta^{(2)}_{v,E}\right]
\end{aligned}$$

,where u, v are respectively the horizontal and vertical velocity components, N, S, E, W is the standard compass convention for 2D grids, q, r are respectively the horizontal and vertical local coordinates (e.g., $q = 0$ on the West face and $q = 1$ on the East face), and $\bar{u}_S, \Delta^{(1)}_{u,S}, \Delta^{(2)}_{u,S}$, etc. are parameters which are determined by an algorithm described in [1].

Some of the important properties of PERM are enumerated below. They indicate why a standard time-stepping scheme, such as RK2, does not perform well with this particular velocity reconstruction. For simplicity, we consider here a grid composed of 2D or 3D cubes, of side Δx.

1. For a given velocity component, in the component direction, the reconstructed field is continuous, piecewise parabolic, and second-order accurate with respect to Δx.

2. The flux of the reconstructed velocity field through any given grid cell face is identically equal to the flux implied by the FV face-normal velocity.

3. For a given velocity component, in the component–normal direction, the reconstructed field is piecewise continuous, piecewise linear, and second-order accurate with respect to Δx.

4. The magnitudes of the velocity discontinuities implied by Property 2 are proportional to Δx^2.

5. The divergence of the reconstructed velocity field is bi- or trilinear, piecewise continuous, and second-order accurate with respect to Δx.

6. The magnitudes of the divergence discontinuities implied by Property 4 are proportional to Δx^2.

It should be noted that, as described in [1], the PERM reconstruction scheme requires a Cartesian grid (either uniform or non-uniform in each coordinate). It is also possible to extend PERM to cylindrical coordinate grids.

Furthermore, we should note from the functional form of PERM as shown above that it is not possible to integrate Eq.(1) analytically-to see why this is so, note that any bi- or tri-linear velocity field is a particular case of a PERM field, and we cannot integrate Eq.(1) exactly for a general bi- or tri-linear velocity field, even if it were time-independent (the differential equation of the Lorentz attractor, for example, is a particular case of a tri-linear velocity field). Therefore, we have to utilize a numerical time-integration scheme for the integration of Eq.(1)–a natural candidate is second-order Runge-Kutta (RK2).

However, second-order time accuracy for the RK2 scheme requires that the velocity field is everywhere continuous, and differentiable (in both space and time) everywhere except on a set of measure zero. Therefore, due to the discontinuities in the PERM reconstructed velocity, an RK2 solution is only first-order accurate in time, for a fixed grid size. Also, applying RK2 to a discontinuous velocity field leads to the violation of an important continuity principle, namely: if two particles are initially infinitesimally close, they remain so. These two statements are demonstrated in the next section.

On the other hand, Items 3 and 5 above imply that for a fine mesh the discontinuities in velocity and divergence, and their detrimental effect on the performance of RK2, are negligible. More specifically, if we decrease both the grid size and time step, keeping the Courant number fixed, a solution which combines PERM and RK2 is second-order accurate. Nevertheless, it is often the case in Large Eddy Simulation/Filtered Density Function (LES/FDF) methods that the grid used provides only marginal resolution of the filtered velocity field, and hence the discontinuities in the PERM reconstructed velocity field should not be neglected.

2 Description of MRK2, and its theoretical advantages over RK2

Let us begin with a simple example of the problems that we encounter when we use RK2 on a discontinuous velocity field. Instead of thinking about a grid, consider an infinite, time-independent velocity field in 2D, which has the following form:

$$U(x,y) = 1 \tag{7}$$

$$V(x,y) = \begin{cases} 0, \text{ for } x < 0 \\ 1, \text{ for } x \geq 0. \end{cases} \tag{8}$$

Here, we can think of the regions $x < 0$ and $x > 0$ as two grid cells, with the region $x = 0$ as the face between them. Note that this velocity field has the kind of discontinuity which can be caused by PERM: U is continuous in the x-direction, and V is continuous in the y-direction, but V is not continuous in the x-direction, its component-normal direction. Now, consider two particles at

time $t = 0$: one at $(X_{1,0}, Y_{1,0}) = (-\Delta t - \epsilon, 0)$ and the other at $(X_{2,0}, Y_{2,0}) = (-\Delta t + \epsilon, 0)$, where $0 < \epsilon \ll \Delta t$. An RK2 step of length Δt has the form:

$$\mathbf{X}^{(1)} = \mathbf{X}_0 + \Delta t\, \mathbf{U}\left(\mathbf{X}_0, 0\right) \tag{9}$$

$$\mathbf{X}^{RK2}_{\Delta t} = \mathbf{X}_0 + \frac{1}{2}\Delta t \left(\mathbf{U}\left(\mathbf{X}_0, 0\right) + \mathbf{U}(\mathbf{X}^{(1)}, \Delta t)\right). \tag{10}$$

Substituting their initial positions for $\mathbf{X}_0$ in Eqs.(9)-(10), we determine that the final positions of the two particles, after one RK2 step of length Δt, are $\left(X^{RK2}_{1,\Delta t}, Y^{RK2}_{1,\Delta t}\right) = (-\epsilon, 0)$ and $\left(X^{RK2}_{2,\Delta t}, Y^{RK2}_{2,\Delta t}\right) = (\epsilon, \Delta t/2)$. On the other hand, with perfect time advancement, the final particle positions at time Δt are $(X_{1,\Delta t}, Y_{1,\Delta t}) = (-\epsilon, 0)$ and $(X_{2,\Delta t}, Y_{2,\Delta t}) = (\epsilon, \epsilon/2)$. Therefore, we can see that the RK2 position of the second particle differs from the correct position by a distance of $(\Delta t - \epsilon)/2$. Keeping in mind that we set $\epsilon \ll \Delta t$, we have that this single time step introduces an $O(\Delta t)$ error in the final position of the second particle. Therefore, RK2 is first order accurate in time overall, and for a time step that crosses a discontinuity it is zeroth-order accurate (in the sense that if all time steps introduced an error of the same magnitude as that introduced by a discontinuity-crossing step, the overall error in the final position of a particle would be $O(\Delta t^0)$).

Furthermore, if we take the limit as $\epsilon \to 0$, we can see that the distance between the initial positions of the two particles goes to 0, but the distance between the final positions after the RK2 step goes to $\Delta t/2$. As previously mentioned, this is a violation of a fundamental continuity principle, namely that if $\mathbf{X}(\mathbf{Y}, t)$ is the Lagrangian mapping between a particle's initial and final positions (i.e. if a particle is at $\mathbf{Y}$ at time 0, it will be at $\mathbf{X}(\mathbf{Y}, t)$ at time t), then $\mathbf{X}(\mathbf{Y}, t)$ is continuous in both $\mathbf{Y}$ and t, except at regions where there is a velocity discontinuity and the face-normal velocity is zero.

The considerable RK2 position error for the second of the two particles is introduced by the following fact: even though under perfect time advancement the second particle is in the region $x > 0$ only for a time $\epsilon \ll \Delta t/2$, the corrected velocity of the RK2 step, $\left(\mathbf{U}\left(\mathbf{X}_{2,0}, 0\right) + \mathbf{U}\left(\mathbf{X}^{(1)}_2, \Delta t\right)\right)/2$, is the average velocity that the particle would experience if it were in that region for a time $\Delta t/2$. In other words, RK2 does not account for how much time a particle spends on each side of a discontinuity. This is the motivation for the Multi-Step RK2 scheme, which we now describe in detail.

MRK2 description

In this section, we present a description of a Multi-Step RK2 time step. We consider a particle that is initially at the position $\mathbf{X}_0$ at time 0, and which is to be advected for a time step of length Δt. For the moment, let us consider the case in which there is only one velocity discontinuity, as in the example above.

First, we take an RK2 predictor step, according to Eq.(9):

$$\mathbf{X}^{(1)} = \mathbf{X}_0 + \Delta t\, \mathbf{U}\left(\mathbf{X}_0, 0\right).$$

If $\mathbf{X}_0$ and $\mathbf{X}^{(1)}$ are in the same cell, then we take the remainder of a standard RK2 time step, according to Eq.(10), which gives the final result

$$\mathbf{X}_{\Delta t}^{MRK2} = \mathbf{X}_{\Delta t}^{RK2} = \mathbf{X}_0 + \Delta t \left(\mathbf{U}\left(\mathbf{X}_0, 0\right) + \mathbf{U}(\mathbf{X}^{(1)}, \Delta t) \right) / 2.$$

Otherwise, let $\mathbf{X}'$ be the point where the ray from $\mathbf{X}_0$ to $\mathbf{X}^{(1)}$ first intersects a cell face. For Cartesian, cylindrical and unstructured polygonal/polyhedronal grids, in order to determine $\mathbf{X}'$ we have to identify the faces of the current grid cell, each of which is a piece of a plane or a cylinder in 2D or 3D, and identify the points where the ray from $\mathbf{X}_0$ to $\mathbf{X}^{(1)}$ crosses each of them, if it does (a simple problem in analytic geometry). Out of these points, $\mathbf{X}'$ is the one closest to $\mathbf{X}_0$.

We make the following second-order accurate estimate for the time, Δt_1, that it takes for the particle to reach this discontinuity:

$$\Delta t_1 = \Delta t \|\mathbf{X}' - \mathbf{X}_0\| / \|\mathbf{X}^{(1)} - \mathbf{X}_0\|, \tag{11}$$

where $\|\cdot\|$ denotes the 2-norm. Next, we break up the RK2 predictor step into two substeps. The first predictor substep, from time 0 to Δt_1, and from $\mathbf{X}_0$ to $\mathbf{X}'$, accounts for the advection of the particle before it reaches the discontinuity. The second predictor substep, which accounts for advection after the particle crosses the discontinuity, takes the particle from time Δt_1 to Δt and from $\mathbf{X}'$ to $\mathbf{X}^{(2)}$, where $\mathbf{X}^{(2)}$ is given by

$$\mathbf{X}^{(2)} = \mathbf{X}' + (\Delta t - \Delta t_1)\mathbf{U}^{+}\left(\mathbf{X}', 0\right). \tag{12}$$

There are two possible definitions of velocity at $\mathbf{X}'$, where the velocity field is discontinuous. The value $\mathbf{U}^{+}\left(\mathbf{X}', 0\right)$ denotes the velocity experienced by the particle after it has crossed the discontinuity. Similarly, below we use $\mathbf{U}^{-}\left(\mathbf{X}', 0\right)$ to denote the velocity experienced by the particle before it crosses a discontinuity (we have assumed, for the moment, that the particle does not cross another discontinuity between $\mathbf{X}'$ and $\mathbf{X}^{(2)}$.)

Next, we make second-order linear approximations in time to estimate the velocities at $\mathbf{X}'$ at the intermediate time Δt_1

$$\begin{aligned} \mathbf{U}^{-}\left(\mathbf{X}', \Delta t_1\right) \approx \hat{\mathbf{U}}^{-}\left(\mathbf{X}', \Delta t_1\right) &= \left[(\Delta t - \Delta t_1)/\Delta t\right] \mathbf{U}^{-}\left(\mathbf{X}', 0\right) \\ &\quad + \left[\Delta t_1/\Delta t\right] \mathbf{U}^{-}\left(\mathbf{X}', \Delta t\right) \end{aligned} \tag{13}$$

$$\begin{aligned} \mathbf{U}^{+}\left(\mathbf{X}', \Delta t_1\right) \approx \hat{\mathbf{U}}^{+}\left(\mathbf{X}', \Delta t_1\right) &= \left[(\Delta t - \Delta t_1)/\Delta t\right] \mathbf{U}^{+}\left(\mathbf{X}', 0\right) \\ &\quad + \left[\Delta t_1/\Delta t\right] \mathbf{U}^{+}\left(\mathbf{X}', \Delta t\right). \end{aligned} \tag{14}$$

Finally, we calculate the corrected velocities for each of the two substeps, and obtain the final particle position

$$\begin{aligned}\mathbf{X}_{\Delta t}^{MRK2} &= \mathbf{X}_0 + (\Delta t_1/2)\left(\mathbf{U}\left(\mathbf{X}_0, 0\right) + \hat{\mathbf{U}}^{-}\left(\mathbf{X}', \Delta t_1\right)\right) \\ &+ \left((\Delta t - \Delta t_1)/2\right)\left(\hat{\mathbf{U}}^{+}\left(\mathbf{X}', \Delta t_1\right) + \mathbf{U}\left(\mathbf{X}^{(2)}, \Delta t\right)\right). \qquad (15)\end{aligned}$$

As already mentioned, for a particle which crosses the discontinuity, we break up the time step into two substeps. The two terms added to $\mathbf{X}_0$ in Eq.(15) are respectively the contributions of each of those two substeps to the advancement of the particle. For simplicity's sake, the algorithm presented above allows for two substeps at most, and can therefore deal with at most one velocity discontinuity crossed by a particle during a time step. By allowing for a greater number of substeps (e.g., breaking up the substep from Δt_1 to Δt into two substeps if the ray from $\mathbf{X}'$ to $\mathbf{X}^{(2)}$ crosses another discontinuity), the algorithm is extended to deal with an arbitrary number of discontinuities crossed by a particle in one time step.

Below, we make a few important observations about MRK2:

1. The MRK2 procedure is almost the same as taking two RK2 steps - one from 0 to Δt_1 and one from Δt_1 to Δt - with the one difference being that the MRK2 procedure does not require knowledge of the velocity field at the intermediate time Δt_1, and uses instead a linear interpolation in time between the velocity fields at time 0 and Δt. This difference is essential in an LES/FDF context, because during each time step a fraction, approximately equal to the Courant number, of the total number of particles in the domain cross a cell face. For a typical LES/FDF calculation with 10^7 particles and a Courant number of 0.1, this means that for each time step there would be approximately 10^6 particles crossing a cell face, each at a different intermediate time Δt_1. It is not computationally feasible to calculate a reconstructed velocity field at each of these intermediate times, because the calculation of the LES FV velocity for one time step, even if we only need the values at the faces of a single grid cell, requires that the density of all the particles is updated, and hence that reaction calculations are performed for each particle. In view of the fact that in a typical LES/FDF calculation over 90% of the computational cost is due to the reaction calculations, this means that calculating the LES FV velocity at intermediate times would increase the overall computational cost by a factor of 10^6.

2. For a fixed grid size, and subject to a condition which is described in the next item, it can be shown analytically that an MRK2 step which crosses a discontinuity introduces an $O(\Delta t^2)$ error in the final particle position, and preserves the continuity of the Lagrangian mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$. Therefore, MRK2 preserves continuity (in a restricted sense) and is second-order accurate in time, since the number of discontinuities crossed by a particle does not increase with decreasing time step.

3. The condition necessary for Item 2 is that the velocity normal to a face (discontinuity) is not identically zero. MRK2 does not preserve the continuity of $\mathbf{X}(\mathbf{Y}, \Delta t)$ where the face-normal velocity is zero. To see this, consider a velocity field similar to the example at the beginning of Section 2, this time with $U(x, y) = 0$, and consider two particles: one at $(X_{1,0}, Y_{1,0}) = (-\epsilon, 0)$ and the other at $(X_{2,0}, Y_{2,0}) = (\epsilon, 0)$. The result of an MRK2 time step is the same as that of an RK2 time step, and of perfect time advancement: $(X_{1,\Delta t}, Y_{1,\Delta t}) = (-\epsilon, 0)$ and $(X_{2,\Delta t}, Y_{2,\Delta t}) = (\epsilon, \Delta t)$. Taking limits as $\epsilon \to 0$, we can see why continuity is violated. Note that this is a property of the particular velocity field considered, not a shortcoming of MRK2 (i.e., even with perfect time advancement there is a continuity violation.)

4. The MRK2 procedure is applicable to a wide variety of grids–it has only two grid-dependent aspects, namely the evaluation of the interpolated velocity at an arbitrary point in space, and the determination of the point $\mathbf{X}'$ where the ray from $\mathbf{X}_0$ to $\mathbf{X}^{(1)}$ first intersects a cell face. The first of these aspects, the ability to evaluate interpolated velocity at an arbitrary point, is a general requirement for and reasonable velocity interpolation scheme. The second aspect, the ability to determine efficiently $\mathbf{X}'$, depends on the grid geometry and, as mentioned in the description of MRK2, reduces to a set of simple analytic geometry problems, such as finding the intersection of a ray with a plane (or a cylinder). Therefore, in addition to Cartesian and cylindrical grids, MRK2 in itself is also applicable to unstructured polygonal/polyhedronal grids. However, we should also keep in mind that the PERM velocity interpolation scheme, which is the primary motivation for the modification from RK2 to MRK2, is currently available only for Cartesian and cylindrical grids.

3 Comparison between the performance of RK2 and MRK2 when applied to a PERM reconstructed velocity field

In this section, we apply both RK2 and MRK2 to a 2D test problem, and compare the performance of the two time-stepping schemes based on three criteria: preservation of continuity, accuracy relative to perfect time advancement, and the maintaining of a consistent subgrid particle distribution.

We use the following problem framework: we have a 2D domain, $[0, 2\pi] \times [0, 2\pi]$, with periodic boundary conditions. We consider two 2D periodic flows. Flow 1 is incompressible, given by the formula

$$U(x, y, t) = 1 - 2\cos(x - t)\sin(y - t), \tag{16}$$
$$V(x, y, t) = 1 + 2\sin(x - t)\cos(y - t). \tag{17}$$

Flow 2 is compressible, given by:

$$U(x,y,t) = 1 - 2\cos(x-t)\sin(y-t) + \frac{1}{2}\sin(x)\cos(2t), \quad (18)$$

$$V(x,y,t) = 1 + 2\sin(x-t)\cos(y-t) + \frac{1}{2}\sin(y)\cos(2t). \quad (19)$$

In order to compute the reconstructed field, the PERM reconstruction scheme uses values of U (according to Eq.(16) or (18)) at the centers of the vertical cell faces, and values of V (according to Eq.(17) or (19)) at the centers of the horizontal cell faces – this simulates FV velocity information. Once the subgrid velocity is reconstructed by PERM, the respective time-advancement scheme is used to advect the particles. The time step Δt is directly proportional to the Courant number (C), which is defined here as:

$$\mathrm{C} = \max\left(\max\left(U(x,y,t)\Delta t/\Delta x\right), \max\left(V(x,y,t)\Delta t/\Delta x\right)\right) \quad (20)$$

where Δx is the side of a grid cell.

3.1 Preservation of continuity, and a qualitative analysis of particle distributions

Here, we present results for the incompressible Flow 1, Eqs. (16) and (17). One of the beneficial properties of PERM is that if the velocity field is divergence free at the FV level, then the subgrid reconstructed velocity is also divergence free. Therefore, applying PERM to Flow 1, we obtain a subgrid velocity field that is divergence free, and so (with perfect time advancement) a particle distribution which is initially uniform remains uniform.

Here, we advect 409,600 particles which are initialized in the following manner: the domain is broken up into 640×640 squares of equal size, and a single particle is placed randomly, with uniform probability density, in each square. This allows for an initial particle distribution which is both very uniform and does not degenerate when a large value of strain is applied to it. Note that the number of particles that we are using is extremely large – 25,600 particles per cell, for a 4×4 grid. Naturally, in practical LES/FDF applications this number is much smaller. The reason why we are using so many here is for the reader to be able to better visualize the mappings $\mathbf{X}(\mathbf{Y}, t)$ between initial and final particle positions. In the following figures, each of the particles is shown as a single green dot, except for those which are initially in the region $[0, \pi/2] \times [0, \pi/2]$, at the lower left corner. These particles have a magenta color, in order to help visualize the strain of the particle distribution.

Figure 2 shows results for a 4×4 grid, with $\mathrm{C} = 1$. We note that this is a rather large Courant number - it is used mainly to emphasize the effects of both time advancement schemes. After a single time step, we can see that the RK2 distribution (top left of Fig. 2) has considerable voids where there are no particles at all - this indicates that the mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$ is not surjective

(onto), since there are values of $\mathbf{X}$ which cannot be attained for any value of $\mathbf{Y}$. We can also see regions in the RK2 distributions where the particles are twice as dense - this indicates that the mapping $\mathbf{X}(\mathbf{Y}, \Delta t)$ is not injective (one-to-one), since regions which were initially separate are mapped on top of each other. This is a serious problem – it introduces error in the subgrid particle distribution (as we will see later on) and violates the fundamental principle that for a periodic flow $\mathbf{X}(\mathbf{Y}, \Delta t)$ is both injective and surjective.

A single MRK2 time step, too, has these problems, but to a much lesser extent – we can see (from the top right of Fig. 2) four elongated gaps where there are no particles, but the size of these gaps is much smaller than the size of the holes in the particle distribution created by RK2. These gaps in the MRK2 distribution are to be expected - they correspond to regions where the face-normal velocity is zero, as explained in section 2. Comparing the particle distributions after one flow-through time (12 time steps), we can see the same differences but with greater magnitude. Whereas MRK2 does not preserve the uniformity of the particle distribution very well, it does much better than RK2, where there are considerably larger void regions.

Figure 3 shows results for the same value of the Courant number, C = 1, but for a finer 8×8 grid. Because the magnitudes of the discontinuities in PERM are $O(\Delta x^2)$ [1], the velocity discontinuities in this case are theoretically smaller by a factor of 4, and RK2 does almost as well as MRK2.

Figure 4 shows results for a 4×4 grid, with a smaller Courant number, C = 0.25. We can again see that RK2 does not preserve continuity very well – there are noticeable voids in the RK2 distribution, both after one time step and after one flow-through time. Although we know theoretically that there are also voids in the MRK2 distributions, these cannot be perceived on the plots. We also note that MRK2 performs better at preserving the sharp interface between the green and purple regions.

Figure 5 shows results for an 8×8 grid, C = 0.25. We cannot perceive a difference between the two particle distributions, which suggests that, due to the smaller value of velocity discontinuities, the performance of RK2 is similar to that of MRK2.

From these comparisons of particle distributions produced by RK2 and MRK2, we conclude that when the velocity discontinuities yielded by the reconstruction are considerable (as for the 4×4 grid), MRK2 performs better than RK2 at preserving the continuity of the particle distribution and the consistency between the subgrid particle distribution and the LES filtered density.

3.2 Accuracy relative to perfect time advancement

As we are focused on developing and demostrating an effective time advancement scheme for a given velocity reconstruction, we are interested in the error of a given solution relative to perfect time advancement. Since we do not have an exact perfect time advancement solution, we use instead an MRK2 solution with a very low Courant number (C = 1/512) as an approximation.

First of all, we consider position error. We use $N = 256$ particles, uniformly distributed across the domain at time 0. We denote by $\mathbf{Y}^{(i)}$ the initial position of the i-th particle, and we denote by $\mathbf{X}^{RK2}\left(\mathbf{Y}^{(i)}, t\right)$, $\mathbf{X}^{MRK2}\left(\mathbf{Y}^{(i)}, t\right)$, and $\mathbf{X}\left(\mathbf{Y}^{(i)}, t\right)$ the corresponding Lagrangian position mappings yielded by RK2, MRK2 and perfect time advancement, respectively. The position error of a given RK2 solution is then defined as

$$\epsilon_x^{RK2} = \frac{1}{N}\left(\sum_{i=1}^{N}\left\|\mathbf{X}^{RK2}\left(\mathbf{Y}^{(i)}, t\right) - \mathbf{X}\left(\mathbf{Y}^{(i)}, t\right)\right\|\right). \tag{21}$$

The MRK2 position error is defined analogously. As we already mentioned, it can be shown analytically that, with Δt being the length of the time step, the RK2 position error is of order $O(\Delta t)$, and the MRK2 position error is of order $O(\Delta t^2)$.

We also consider error of infinitesimal volume expansion (which we will refer to as dV error, for the sake of brevity). Note that we refer here to two-dimensional volume, i.e. area. For an infinitesimal material element whose initial volume is dV_0, and whose initial position is $\mathbf{Y}$, the final volume, dV_t, is given by

$$dV_t = dV_0 \det\left(\frac{\partial X_j(\mathbf{Y}, t)}{\partial Y_k}\right), \tag{22}$$

where $\mathbf{X}(\mathbf{Y}, t)$ is the Lagrangian position mapping previously mentioned. This is the basis of the change of variables formula, which in this context tells us that the volume at time t of a set S is

$$V_t = \int_S \det\left(\frac{\partial X_j(\mathbf{Y}, t)}{\partial Y_k}\right)|d\mathbf{Y}|, \tag{23}$$

provided that the mapping $\mathbf{X}(\mathbf{Y}, t)$ is one-to-one. Therefore, accurate values of $\det\left(\partial X_j/\partial Y_k\right)$ are essential for achieving an accurate subgrid particle distribution. We also note that $\mathbf{X}(\mathbf{Y}, t)$ may not be one-to-one if its continuity is violated (such as in Figure 1, where particles are twice as dense in some places, because certain regions are mapped on top of each other), and therefore continuity of the position mapping, which is considered in the previous subsection, is just as essential for getting the right value of V_t as are accurate values of $\det\left(\partial X_j/\partial Y_k\right)$.

With this in mind, we define dV error of a given RK2 solution as

$$\epsilon_{dV}^{RK2} = \frac{1}{N}\sum_{i=1}^{N}\left|\det\left(\frac{\partial X_j^{RK2}\left(\mathbf{Y}^{(i)}, t\right)}{\partial Y_k}\right) - \det\left(\frac{\partial X_j\left(\mathbf{Y}^{(i)}, t\right)}{\partial Y_k}\right)\right| \tag{24}$$

The MRK2 dV error is similarly defined. For RK2, it can be proven analytically that the dV error is $O(\Delta t^2)$ when the divergence of the PERM reconstructed field is continuous everywhere, and $O(\Delta t)$ when the divergence is discontinuous from cell to cell, which is the case for all compressible flows.

Here, we cannot calculate the necessary Jacobians exactly, but a sufficiently accurate numerical estimate which we use is

$$\det\left(\frac{\partial X_j}{\partial Y_k}\right) \approx A_t/A_0 \tag{25}$$

where A_0 is the initial area of a triangle initially of sides 10^{-8} formed by three particles at time 0, and A_t is the area of the triangle formed by those three particles at time t.

Here, we present results for the compressible Flow 2, Eqs.(18) and (19). Figure 6 shows plots of position error and dV error for 4×4 and 8×8 grid solutions, with the Courant number ranging from 1 down to 1/32. For both position error and dV error, RK2 exhibits first-order behavior, whereas MRK2 exhibits second-order behavior. This is consistent with previously stated analytic results, and it also indicates that for MRK2, dV error is $O(\Delta t^2)$ - a result which we have not been able to prove rigorously.

It should also be noted that, for larger values of C, MRK2 is computationally slower than RK2. In order to compare the efficiency of MRK2 with that of RK2, Figure 7 shows plots of the position and dV errors from Figure 6 versus simulation time, for a simulation with 1600 particles per cell, instead of just the 256 particles (for the entire domain) that we used to determine the errors. We use a larger number of particles here to ensure that the cost of all the operations other than particle advection (such as the determination of the PERM coefficients) is negligible.

From Figure 7 we see that MRK2 becomes more efficient as the error (and hence the Courant number) decreases–for the 4×4 grid, MRK2 is more efficient for $\mathrm{C} \leq 1/2$, and for the 8×8 grid, MRK2 is more efficient for $\mathrm{C} \leq 1/4$. On the other hand, Figure 7 takes into account only the computational cost for particle advection–if we also consider the cost of chemistry calculations, which is usually much higher than the cost of particle advection, we conclude that MRK2 may be more efficient than RK2 even for larger values of C.

Table 1 shows values of the error, as well as simulation time for a 1600-particles-per-cell simulation, for $\mathrm{C} = 1$ and for $\mathrm{C} = 0.25$. For both values of C, and for both meshes, MRK2 has a smaller position error: $\epsilon_X^{MRK2}/\epsilon_X^{RK2} < 0.69$. For both meshes, the MRK2 dV error is less than the RK2 dV error for $\mathrm{C} = 0.25$, but for $\mathrm{C} = 1$ and the 8×8 mesh, the MRK2 dV error is greater than the RK2 dV error. This, however, is not a major concern, because as we already noted, $\mathrm{C} = 1$ is a very large value for the Courant number - in most applications, we expect a smaller value, such as 0.25, to be used.

3.3 Accuracy of the subgrid particle distribution

Previously, McDermott and Pope [1] have shown that a PERM reconstructed velocity field, with RK2 as the time-stepping scheme, performs much better than linear interpolation in maintaining a subgrid particle distribution which is consistent with the LES filtered density. Here, we demonstrate that this consistency becomes even better when MRK2 is used as the time-stepping scheme.

For the present tests, we use the incompressible Flow 1, Eqs.(16) and (17). The particles are initialized in a manner similar to that in Section 3.1 (by breaking up the domain into as many squares as there are particles, and initializing a single particle randomly into each square). Then the mean particle mass density is initially the same everywhere in the flow, and it remains so under perfect time advancement, for a divergence-free velocity field such as the one we consider. Hence, the expected number of particles in a set S is directly proportional to the area of S.

We quantify the accuracy of the subgrid particle distribution in the following way: we break up the domain into equally sized squares, which we call sampling squares. We use sampling squares of three different sizes, with sides $\pi/4$, $\pi/8$, and $\pi/16$ (respectively 1/2, 1/4 and 1/8 of the sidelength of a grid cell, for a 4×4 grid). At any given timestep, let N_i be the number of particles in the i-th sampling square, let N_{mean} be the expected number of particles for a square of this size, and let N_{sq} be the number of sampling squares in the domain. The subgrid particle distribution error is then defined as:

$$\epsilon_{SG} = \left(\sum_{i=1}^{N_{sq}} \left| \left(N_i - N_{mean} \right) / N_{mean} \right| \right) / N_{sq}. \tag{26}$$

This subgrid distribution error has two contributions: first, a probabilistic error, due to the random particle initialization; and, second, a bias error, due to the particle advection scheme (an example of this are the considerable voids in the particle distribution left by RK2 at C = 1). For this test flow, the PERM reconstructed velocity field has zero divergence, and therefore a perfect time advancement solution contains only the probabilistic component of the error. Therefore, we compare the subgrid distribution errors given by the MRK2, RK2 and perfect time advancement solutions, making the assumption that whatever is in excess of the perfect time advancement error is mostly bias error.

Figure 8 shows results for 65,536 particles, C = 0.25 and a 4×4 grid. As may be seen, the MRK2 subgrid distribution error is much closer to the perfect time advancement error than is the RK2 error. From this we deduce that, for the coarse 4×4 grid, the bias error caused by MRK2 is smaller than the bias error caused by RK2.

Figure 9 shows results for 65,536 particles, C = 0.5 and a 8×8 grid. The time step is the same as for the results on Figure 6, but the velocity field is better resolved by PERM. Here, the MRK2, RK2 and perfect time advancement errors are much closer together, and MRK2 does not provide a big advantage over RK2. Again, we observe that for a finer grid RK2 does not perform considerably worse than MRK2, because the discontinuities in velocity are smaller.

3.4 Computational cost

We have seen that MRK2 has superior accuracy than RK2, and has better performance in preserving the consistency between the subgrid particle distribution and the filtered LES density. One disadvantage that MRK2 has over RK2 is

that it is slower–this was mentioned in Section 3.2, where we compared the efficiency of MRK2 with that of RK2.

For the well-optimized Fortran 90 implementation of MRK2 which we used to obtain the results in the previous sections, each additional substep after the first one costs approximately 1.5 times the cost of an RK2 step. In other words, if the total time for one RK2 time step for one particle is t_{step}, the total time for an MRK2 time step, for a particle which crosses n discontinuities, is approximately $t_{step}(1+1.5n)$. Therefore, for a flow in which each particle crosses one discontinuity per time step (hence requiring 2 substeps per time step) an MRK2 calculation would require 2.5 times more time than an RK2 calculation would.

Fortunately, in most flows of practical interest the number of particles which require two or more substeps is not quite so high. For the test cases from the previous sections, the MRK2 calculations have a 70% computational overhead at C = 1, and a 20% computational overhead at C = 0.25, relative to the RK2 calculations. For smaller values of C the computational overhead is even smaller, since the ratio between the number of particles which cross a discontinuity and the number of particles which do not cross one decreases with decreasing Courant number.

4 Conclusions

We have outlined a new time-stepping scheme, based on second-order Runge-Kutta, which is particularly suited for advection of a large number of particles through a discontinuous velocity field, such as the one yielded by the Parabolic Edge Reconstruction Method (PERM). We have demonstrated that the new scheme, MRK2, preserves better than RK2 the continuity of the Lagrangian mapping between initial and final positions. We have also shown that MRK2 is second-order accurate in time, as opposed to RK2, which is first-order accurate for a discontinuous velocity field. Additionally, we have shown that MRK2 preserves better than RK2 the consistency between the subgrid particle distribution and the LES filtered density.

On the other hand, we have seen that the advantages of MRK2 over RK2 diminish when a more refined grid is used for velocity reconstruction, and that MRK2 has a computational overhead relative to RK2 (20% at C = 0.25) which is not negligible. Therefore, whereas RK2 is sufficiently accurate (and faster) for problems with a fine mesh, it is preferable to use MRK2 in problems where the mesh provides only marginal resolution of the velocity, which is often the case in LES/FDF methods.

Finally, it should be noted that the main principle of MRK2 - breaking up a step into two or more substeps every time a velocity discontinuity is encountered - can be applied in a straightforward manner to time advancement schemes other than RK2. The Midpoint Euler scheme, for example, can be modified to deal with velocity discontinuities similarly to the way MRK2 does. This would be preferable in a situation where velocity information is known at the middle of

the time step.

Acknowledgements

This work is supported by the Air Force Office of Scientific Research, Grant FA9550-06-1-0048.

References

[1] R. McDermott, and S. B. Pope, *The Parabolic Edge Reconstruction Method (PERM) for Lagrangian Particle Advection*, Submitted to the Journal of Computational Physics

[2] M. Muradoglu, S. B. Pope, and D. A. Caughey, *The Hybrid Method for the PDF Equations of Turbulent Reactive Flows: Consistency Conditions and Correction Algorithms*, Journal of Computational Physics **172**, 841–878 (2001)

[3] P. Jenny, S. B. Pope, M. Muradoglu, and D. A. Caughey, *A Hybrid Algorithm for the Joint PDF Equation of Turbulent Reactive Flows*, Journal of Computational Physics **166**, 218–252 (2001)

[4] Y. Z. Zhang, D. C. Haworth, *A General Mass Consistency Algorithm for Hybrid Particle/Finite-Volume PDF Methods*, Journal of Computational Physics **194**, 156–193 (2004)

[5] S. B. Pope, *Turbulent Flows*, (Cambridge Univ. Press, Cambridge, UK, 2000)

[6] D. E. Kincaid, E. W. Cheney, *Numerical Analysis: Mathematics of Scientific Computing* , Brooks Cole; 3 edition (October 25, 2001)

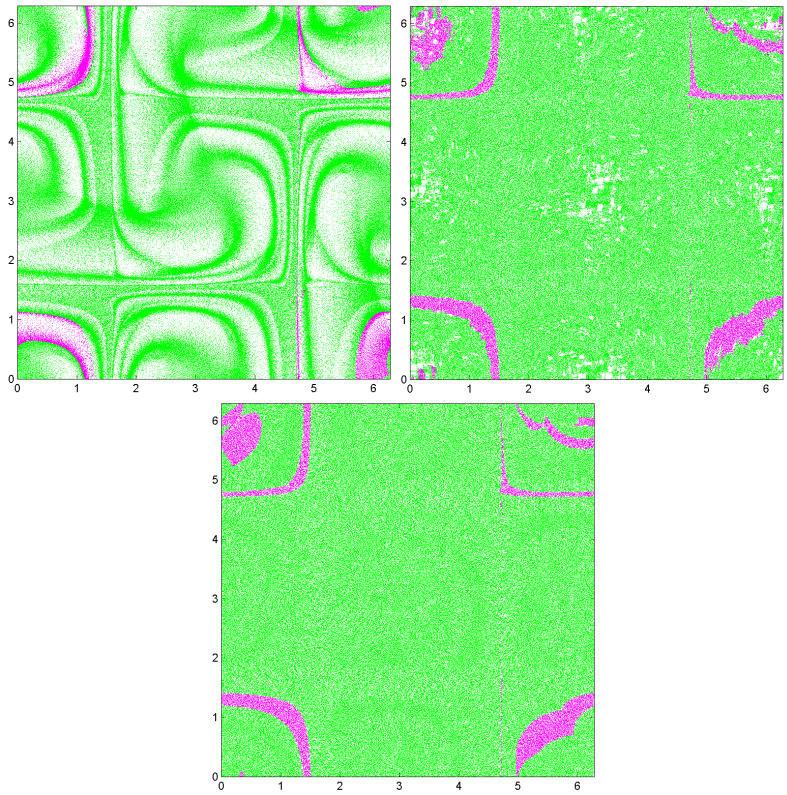

Figure 1: Comparison between the performance of the bilinear and PERM velocity interpolation schemes, and between the RK2 and MRK2 time-stepping schemes. The test flow is incompressible and the initial particle distribution is uniform. Under these conditions, the final particle distribution should be uniform. Top left: final particle distribution yielded by bilinear interpolation and RK2 time-stepping, top right: final particle distribution yielded by PERM interpolation and RK2 time-stepping, bottom: final particle distribution yielded by PERM interpolation and MRK2 time-stepping.

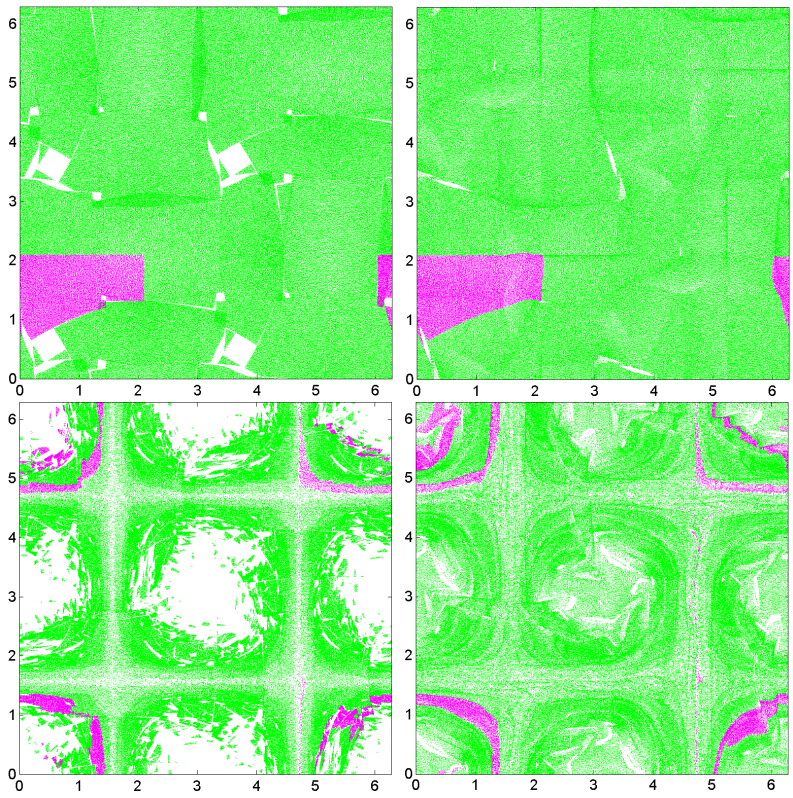

Figure 2: Particle distributions in Flow 1 for a 4×4 grid, CFL $= 1$. Top left: RK2, one time step ($\Delta t = 0.448$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (12 time steps); bottom right: MRK2, one flow-through time. The magenta-colored particles are those that are in the lower-left corner of the domain at $t = 0$.

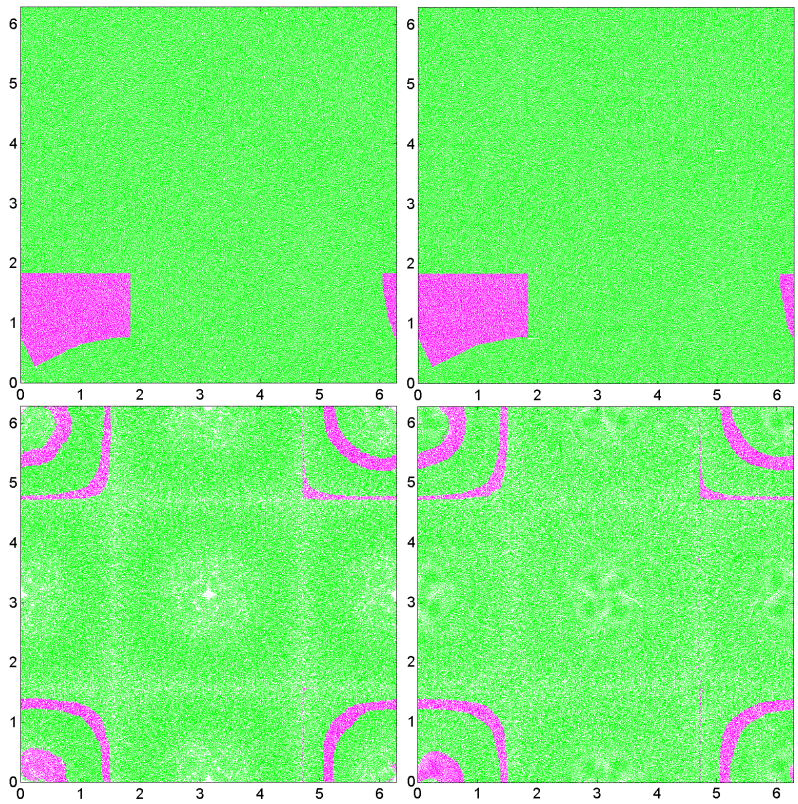

Figure 3: Particle distributions in Flow 1 for an 8×8 grid, CFL $= 1$. Top left: RK2, one time step ($\Delta t = 0.224$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (24 time steps); bottom right: MRK2, one flow-through time. The magenta-colored particles are those that are in the lower-left corner of the domain at $t = 0$.

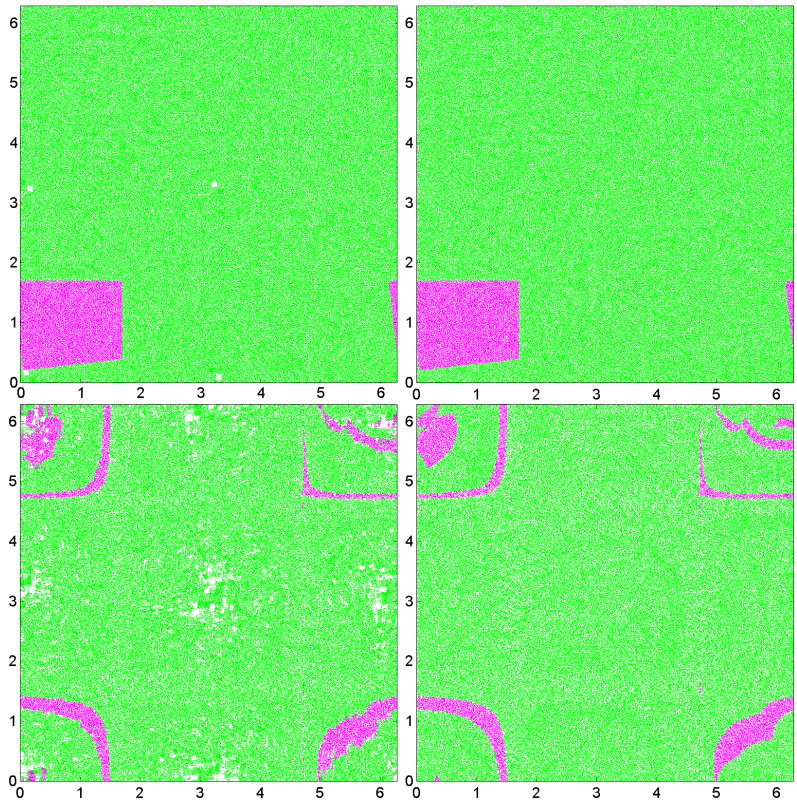

Figure 4: Particle distributions in Flow 1 for a 4×4 grid, CFL $= 0.25$. Top left: RK2, one time step ($\Delta t = 0.112$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (48 time steps); bottom right: MRK2, one flow-through time. The magenta-colored particles are those that are in the lower-left corner of the domain at $t = 0$.

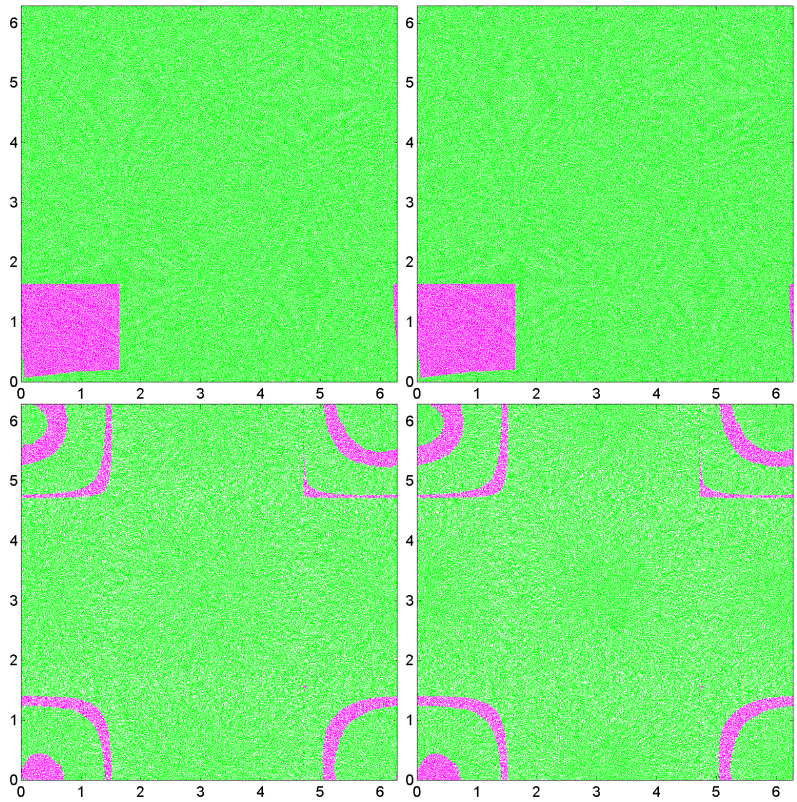

Figure 5: Particle distributions in Flow 1 for an 8×8 grid, CFL $= 0.25$. Top left: RK2, one time step($\Delta t = 0.056$); top right: MRK2, one time step; bottom left: RK2, one flow-through time (96 time steps); bottom right: MRK2, one flow-through time. The magenta-colored particles are those that are in the lower-left corner of the domain at $t = 0$.

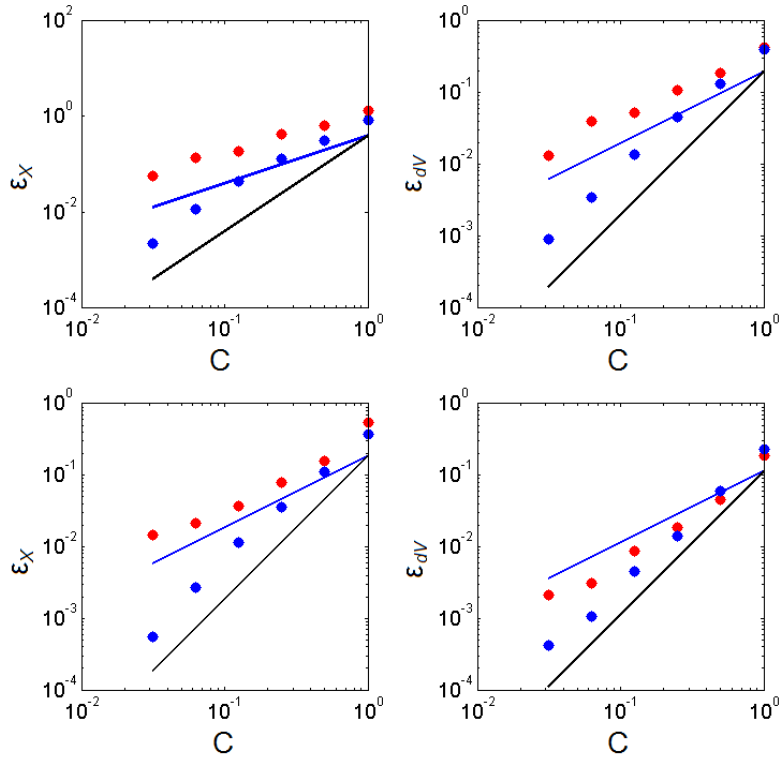

Figure 6: Results for position and dV error in Flow 2 at $t = 8.97$. Top left: position error for a 4×4 grid; top right: dV error for a 4×4 grid; bottom left: position error for a 8×8 grid; bottom right: dV error for a 8×8 grid. The red circles denote RK2 error, the blue circles denote MRK2 error. The blue solid line indicates first-order convergence, and the black solid line indicates second-order convergence.

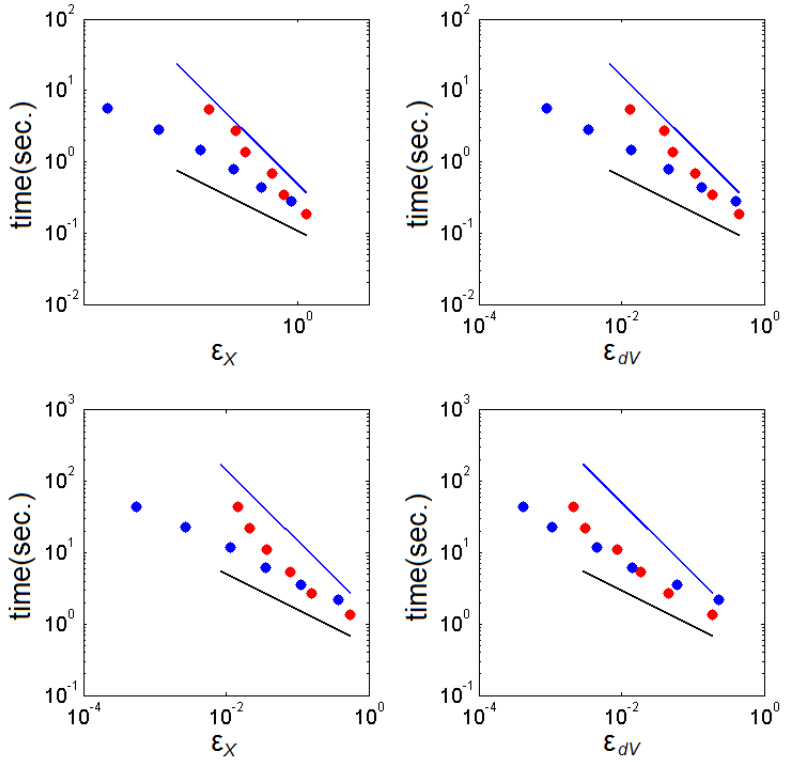

Figure 7: Comparison between the efficiency of MRK2 and RK2, based on position and dV error, and computational cost of particle advection for a 1600-particles-per-cell simulation of Flow 2 at $t = 8.97$. Top left: position error for a 4×4 grid; top right: dV error for a 4×4 grid; bottom left: position error for a 8×8 grid; bottom right: dV error for a 8×8 grid. The red circles denote RK2 error, the blue circles denote MRK2 error. The blue solid line indicates first-order convergence, and the black solid line indicates second-order convergence.

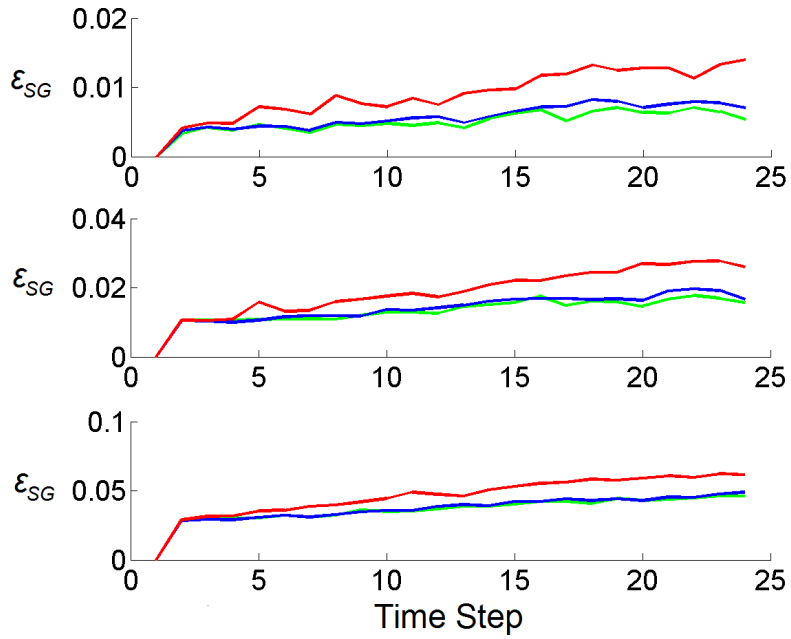

Figure 8: Results for subgrid particle distribution error for Flow 1. CFL = 0.25, ($\Delta t = 0.112$), 4×4 grid Top: error for sampling squares of side $\pi/4$; middle: error for sampling squares of side $\pi/8$; bottom: error for sampling squares of side $\pi/16$. The errors for the RK2, perfect time advancement, and MRK2 solutions are shown respectively by the red, green, and blue curves.

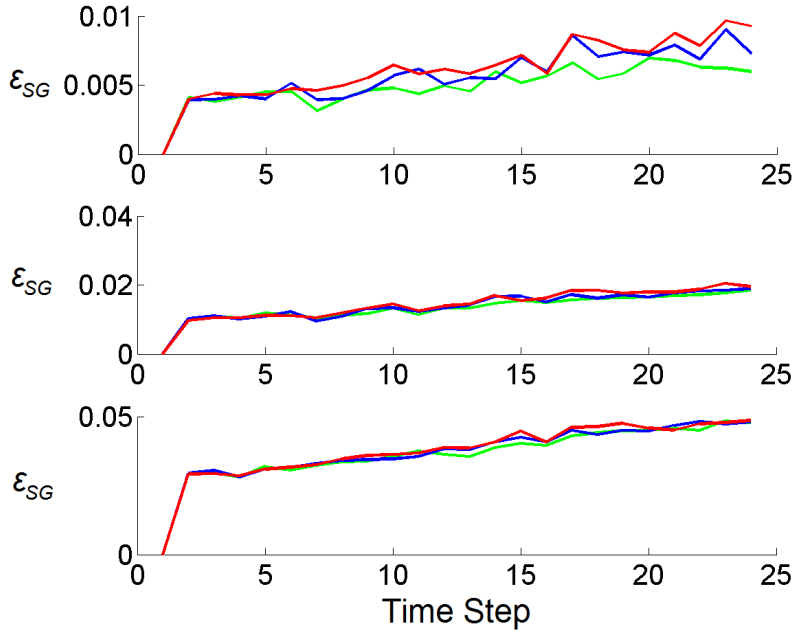

Figure 9: Results for subgrid particle distribution error for Flow 1. CFL = 0.5, ($\Delta t = 0.112$), 8×8 grid Top: error for sampling squares of side $\pi/4$; middle: error for sampling squares of side $\pi/8$; bottom: error for sampling squares of side $\pi/16$. The errors for the RK2, perfect time advancement, and MRK2 solutions are shown respectively by the red, green, and blue curves.

The Direct Richardson p-th Order (DRp) Schemes - A New Class of Time Integration Schemes for Stochastic Differential Equations

Pavel P. Popov
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

Stephen B. Pope
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

October 03, 2011

Abstract

We describe a new family of weak p-th order accurate SDE time integration schemes, called the Direct Richardson p-th order accurate (DRp) schemes. The DRp schemes use the idea of Richardson extrapolation on Euler time steps, performed by way of an acceptance-rejection algorithm. Previous applications of Richardson extrapolation to the Euler scheme are applicable only when the objective is to estimate a functional of the final distribution of the process. In contrast, provided that the diffusion matrix is strictly positive definite, the DRp class of schemes can be used in all applications which require a weak SDE time integration scheme. Numerical results have been obtained, and a comparison is made between the second- and third-order accurate DRp schemes and other modern SDE time integration schemes, indicating that the DRp schemes incur less error than standard algorithms based on Ito-Taylor expansions, and have similar computational efficiency. Finally, we provide a proof of the convergence properties of the DRp schemes.

AMS subject classifications: 60H35, 65C30

1 Introduction

In this work, we introduce a new class of weak p-th order accurate numerical schemes for the integration of the n-dimensional non-homogeneous and

anisotropic Ito stochastic differential equation (SDE):

$$\mathrm{d}\mathbf{X} = \mathbf{D}\left(\mathbf{X}, t\right)\mathrm{d}t + \sigma\left(\mathbf{X}, t\right)\mathrm{d}\mathbf{W}, \tag{1}$$

where $\mathbf{X}(t)$ is the random process governed by the Ito SDE, $\mathbf{D}\left(\mathbf{x}, t\right)$ denotes the drift, which is an n-dimensional vector field, $\sigma\left(\mathbf{x}, t\right)$ is a strictly positive definite $n \times n$ matrix field, and $\mathrm{d}\mathbf{W}$ as usual indicates that the SDE is driven by the standard n-dimensional Wiener process. In this form and generality, the Ito SDE has numerous applications in science and engineering [4]: for example, its solution is an essential component of particle-based numerical schemes for turbulent combustion [3], which is the authors' particular interest.

Given the wide range of applicability of the Ito SDE, it is not suprising that over the years numerous methods have been developed for its numerical approximation, in both the strong and weak senses. Strong SDE integration schemes aim to accurately reconstruct the trajectory $\mathbf{X}\left(t\right)$ as a function of the Wiener sample path $\mathbf{W}\left(t\right)$. In contrast, weak schemes only need to satisfy the condition that the distribution of the numerical solution approximates that of the actual SDE solution. Here, we concentrate on numerical schemes which exhibit weak convergence.

One of the most widespread numerical schemes for the solution of non-stiff SDEs in Ito form is the forward Euler scheme. Using Δt to denote the length of a time step, and η to denote a sample vector from the n-dimensional standard normal distribution $\mathcal{N}\left(\mathbf{0}, I\right)$, a single time step of the Euler scheme has the form

$$\mathbf{Z}^E\left(t + \Delta t\right) = \mathbf{Z}^E\left(t\right) + \mathbf{D}\left(\mathbf{Z}^E\left(t\right), t\right)\Delta t + \sigma\left(\mathbf{Z}^E\left(t\right), t\right)\eta\sqrt{\Delta t}, \tag{2}$$

and it is well-known [4] that the Euler scheme is strong 0.5^{th} order accurate and weak 1^{st} order accurate.

There are also numerous higher-order weak SDE integration schemes in existence [7,4,3,12]. To the authors' knowledge, all of these are based on approximating higher-order terms in the Ito-Taylor expansion of $\mathbf{X}\left(t\right)$. One way to perform this approximation is through the explicit evaluation of derivatives of the fields $\mathbf{D}\left(\mathbf{x}, t\right)$ and $\sigma\left(\mathbf{x}, t\right)$ at the initial point (for an example, see the second-order weak Taylor scheme presented in [4]). Alternatively, the fields can be sampled at additional points, thereby implicitly obtaining the required derivatives through finite differences (for an example, see the two families of weak second-order derivative-free schemes developed in [12]).

Here, we present a different approach - to approximate each time step increment through Euler steps of varying length, and then employ Richardson extrapolation via an acceptance/rejection procedure in order to obtain a higher-order scheme. This may sound somewhat familiar to the reader, inasmuch as Richardson extrapolation on Euler solutions is a well-known [9] method for obtaining higher-order estimates of functionals $E\left(g\left(\mathbf{X}(T)\right)\right)$ of the solution at the end time T. Additionally, the recently developed multilevel Monte Carlo method [5,6] uses an approximation based on Euler solutions with varying time steps to minimize the computational cost for approximating functionals of $\mathbf{X}(t)$ over

the entire interval $t \in [0, T]$. Furthermore, the acceptance/rejection methodology has previously been used in SDE time-stepping algorithms, for example the method introduced in [2], for the purposes of obtaining an integrator which is ergodic with respect to the SDE equilibrium distribution.

The significant difference in the class of schemes that we present is that the Richardson extrapolation is applied at each time step directly to the PDF of the random variable that approximates the SDE solution. This yields a family of SDE integration schemes, called the Direct Richardson p-th Order (DRp) schemes, which are p-th order accurate at each time step, and not just at the end of the simulation. This high-order accuracy at each time step is essential when the solution to the SDE is just one part of a more complex simulation, such as for example in a Lagrangian Monte Carlo solution for turbulent reactive flows.

To elaborate on this issue, we note that in such a solution [13], the chemical composition in the reaction domain is represented by an ensemble of particles whose temperature and density affects the velocity and diffusivity of the flow. The solution algorithm alternates between an SDE time step, which advances the particle locations using the current velocity and diffusivity fields, and a Finite-Volume time step, which solves the Navier-Stokes and turbulence modeling equations, using density and temperature fields based on the new particle locations.

Thus, while the methods described in [9,5,6] are computationally more efficient for the purpose of evaluating expectations of functionals of an a priori known SDE, they cannot be employed in an application of mathematical physics such as the one outlined in the above paragraph, because in such an application the coefficients of eq.(1) at a given time t are not known until the particles' positions have been advanced to time t. For this reason, we compare the performance of the DRp class of schemes with that of SDE time integrators such as those of Kloeden and Platen [4] and the Cao and Pope [3], which can also be used in an application where the SDE coefficients are known only locally in time, i.e., only for the duration of the current finite volume time step.

In this work, we present the general form of the DRp family of SDE integration schemes and provide a proof of the weak p-th order accuracy of its members, subject to certain smoothness criteria on the SDE drift and diffusion terms. We also provide results from numerical test cases which compare the performance of the two simplest DRp schemes - DR2 and DR3 - with that of other modern weak second-order accurate SDE integration schemes.

The organization of the rest of this paper is as follows: in Section 2, we present the simplest member of the DRp family - DR2 - and comment on its implementation in a computational code. In Section 3, we present results from numerical test cases which compare the performance of DR2 and DR3 with that of two modern weak second-order schemes. The numerical results indicate that, for the purpose of approximating Ito SDEs with strictly positive definite diffusion, the DRp schemes are at least as efficient as current SDE integration schemes.

Section 4 introduces Richardson extrapolation and the way in which it is

employed in the DRp schemes. Section 5 introduces the regularity and boundedness criteria which need to be satisfied by the fields $\mathbf{D}(\mathbf{x},t)$ and $\sigma(\mathbf{x},t)$ in order to achieve weak p-th order accuracy. Section 6 introduces the framework and notation for the general form of the DRp schemes, and Section 7 gives a pseudocode description of the general DRp scheme. Section 8 contains a proof of the weak p-th order accuracy of the DRp schemes, and Section 9 provides a summary. Finally, Appendices 1 and 2 contain the proofs of two theorems which are used in Section 8.

The mathematical developments in the second half of this paper are essential, as they prove the convergence of the schemes proposed by the authors. Nevertheless, it is appreciated that some readers are primarily interested in employing a DRp scheme for a particular application, and may wish to skip the more theoretical aspects of this work. Such readers are advised to read Sections 2, 3, 5, as well as Section 4 up to and including eq.(20), and Section 7 up to and including eq.(41).

2 Description of the Second-Order Member of the DRp Family - DR2

Below, we give a pseudocode description of a single time step of length Δt of the weak second-order accurate DR2 scheme. Without loss of generality, we assume that the initial position is $\mathbf{X}(t=0)=\mathbf{0}$. Also, for the sake of compactness of notation, we shall use $B(\mathbf{x},t)$ to denote $\sigma(\mathbf{x},t)\,\sigma^T(\mathbf{x},t)$.

1. Obtain two independent samples $\eta_1,\eta_2\sim\mathcal{N}(\mathbf{0},I)$ from the standard normal distribution.

2. Set

$$\begin{aligned}\mathbf{U}_1^{(2)} &\equiv \mathbf{0}+\mathbf{D}(\mathbf{0},0)\frac{\Delta t}{2}+\sigma(\mathbf{0},0)\,\eta_1\sqrt{\frac{\Delta t}{2}}\\ \mathbf{U}_2^{(2)} &\equiv \mathbf{U}_1^{(2)}+\mathbf{D}\left(\mathbf{U}_1^{(2)},\frac{\Delta t}{2}\right)\frac{\Delta t}{2}+\sigma\left(\mathbf{U}_1^{(2)},\frac{\Delta t}{2}\right)\eta_2\sqrt{\frac{\Delta t}{2}}.\end{aligned}\quad(3)$$

3. Compute

$$\begin{aligned}\mathbf{v}_1 &\equiv \mathbf{U}_2^{(2)}-\mathbf{U}_1^{(2)}-\mathbf{D}(\mathbf{0},0)\frac{\Delta t}{2}\\ \mathbf{v}_2 &\equiv \mathbf{U}_2^{(2)}-\mathbf{U}_1^{(2)}-\mathbf{D}\left(\mathbf{U}_1^{(2)},\frac{\Delta t}{2}\right)\frac{\Delta t}{2}\\ f_1 &\equiv \frac{1}{|B(\mathbf{0},0)|^{1/2}}\exp\left(-\frac{1}{\Delta t}\mathbf{v}_1^T\left(B(\mathbf{0},0)\right)^{-1}\mathbf{v}_1\right)\end{aligned}$$

$$
\begin{aligned}
f_2 &\equiv \frac{1}{\left|B\left(\mathbf{U}_1^{(2)}, \frac{\Delta t}{2}\right)\right|^{1/2}} \exp\left(-\frac{1}{\Delta t}\mathbf{v}_2^T\left(B\left(\mathbf{U}_1^{(2)}, \frac{\Delta t}{2}\right)\right)^{-1}\mathbf{v}_2\right) \\
H &\equiv 1 - \frac{f_1}{2f_2}
\end{aligned} \tag{4}
$$

4. Sample a 1D random variable ξ from the standard uniform distribution: $\xi \sim U(0,1)$

5. If the following acceptance criterion is met

$$
\xi < \max(H, 0.1), \tag{5}
$$

then set $\mathbf{Z}_{\Delta t}^{DR2} = \mathbf{U}_2^{(2)}$. Else, go back to step 1.

The above algorithm is easily implemented in a computational code - each acceptance-rejection step involves two evaluations of the drift and diffusion fields, along with the computations involved in calculating H, which are trivial for the case of isotropic diffusion, and can be optimized to involve just one determinant evaluation and one solution of an $n \times n$ linear system for the anisotropic case.

Taking a more abstract view on the DR2 algorithm, we can see that it is basically a rejection sampling algorithm which takes two Euler time steps (eq.(3)) at each sampling step, and accepts with probability $\max(H, 0.1)$. The reader is referred to [1] for an accessable introduction to rejection sampling methods, also known as acceptance-rejection methods. The essential idea of acceptance-rejection methods is that we can generate a random variable with a desired distribution (the target distribution), by obtaining a sample from a simpler distribution on the same sample space (the instrumental distribution), and accepting that sample with a probability based on the ratio of the PDFs associated with the target and instrumental distributions.

One might wonder, then, where exactly is the Richardson extrapolation after which this scheme is named. To answer this question, let us define $f_{\Delta t}^2(\mathbf{x}, \mathbf{y})$ to be the PDF of the random variable $\left(\mathbf{U}_1^{(2)}, \mathbf{U}_2^{(2)}\right)$, in other words the PDF associated with the event $\left\{\mathbf{U}_1^{(2)} = \mathbf{x}, \mathbf{U}_2^{(2)} = \mathbf{y}\right\}$. Also, let $\eta_3 \sim \mathcal{N}(\mathbf{0}, I)$, define $\overline{\mathbf{U}_2^{(2)}}$ by

$$
\overline{\mathbf{U}_2^{(2)}} \equiv \mathbf{U}_1^{(2)} + \mathbf{D}(\mathbf{0}, 0)\frac{\Delta t}{2} + \sigma(\mathbf{0}, 0)\eta_3\sqrt{\frac{\Delta t}{2}}, \tag{6}
$$

and let us define $f_{\Delta t}^1(\mathbf{x}, \mathbf{y})$ to be the PDF of the random variable $\left(\mathbf{U}_1^{(2)}, \overline{\mathbf{U}_2^{(2)}}\right)$, in other words the PDF associated with the event $\left\{\mathbf{U}_1^{(2)} = \mathbf{x}, \overline{\mathbf{U}_2^{(2)}} = \mathbf{y}\right\}$. It can then be shown that

$$\frac{f^1_{\Delta t}\left(\mathbf{U}^{(2)}_1, \mathbf{U}^{(2)}_2\right)}{f^2_{\Delta t}\left(\mathbf{U}^{(2)}_1, \mathbf{U}^{(2)}_2\right)} = \frac{f_1}{f_2}, \tag{7}$$

and so, since $f^2_{\Delta t}(\mathbf{x}, \mathbf{y})$ is the instrumental distribution of the rejection sampling, we arrive at the conclusion that the PDF of the pair $\left(\mathbf{U}^{(2)}_1, \mathbf{U}^{(2)}_2\right)$ which gets accepted by the DR2 algorithm, approximates $2f^2_{\Delta t}(\mathbf{x}, \mathbf{y})\left(1 - \frac{f^1_{\Delta t}(\mathbf{x},\mathbf{y})}{2f^2_{\Delta t}(\mathbf{x},\mathbf{y})}\right) = 2f^2_{\Delta t}(\mathbf{x}, \mathbf{y}) - f^1_{\Delta t}(\mathbf{x}, \mathbf{y})$. For the above result to hold, however, we need to show that the probability of H deviating significantly from $\frac{1}{2}$ is negligible - we shall elaborate on this claim, and demonstrate its correctness, later on.

Note, however, that eq.(6) implies that $\overline{\mathbf{U}^{(2)}_2}$ has the same distribution as the end value after a single Euler step of length Δt, whereas by its definition, $\mathbf{U}^{(2)}_2$ is the end value after two Euler steps of length $\frac{\Delta t}{2}$ each. Therefore, the PDF of $\mathbf{Z}^{DR2}_{\Delta t}$ approximates the second order Richardson extrapolation of the PDFs of the final value at $t = \Delta t$, as given by one and two Euler steps.

And so, we see that the DR2 scheme applies Richardson extrapolation directly to the PDF of the process' value after a single time step. This makes it weakly second-order accurate, as demonstrated by the numerical results presented in the next section.

3 Numerical Test Cases - Comparing the Accuracy and Efficiency of DR2 and DR3 with that of Other Weakly Convergent SDE Integration Schemes

In this section, we provide results from numerical test cases which compare the accuracy and computational cost of the DR2 and DR3 schemes with those of two other modern SDE time integration schemes. The first of these is the weak second-order midpoint scheme, developed by Cao and Pope [3], which we refer to as the Cao and Pope scheme, or CP for short. In the authors' experience [8], the Cao and Pope SDE integration scheme is both accurate and computationally efficient. However, having been designed specifically for its application in Lagrangian Monte Carlo methods for turbulent combustion, the CP scheme can only treat cases with isotropic diffusion. We compare its performance with that of DR3 on a one-dimensional, and hence isotropic test case.

We also make a comparison with the multi-dimensional explicit second-order weak scheme described by Kloeden and Platen [4, p. 486-487] and generalized to a family of derivative-free weak second-order schemes by Tocino and Vigo-Aguiar [12] (the particular member of the family of schemes used here is referred to as SIE-A in [12]). As the Kloeden and Platen (KP) scheme allows

for anisotropic diffusion, we compare its performance with that of DR2 on a two-dimensional anisotropic test case.

Two dimensional, anisotropic test case - comparison between the DR2 and KP schemes

We perform a simulation on the domain $\mathbf{x} = (x, y) \in [0, 2\pi) \times [0, 2\pi)$ with periodic boundary conditions, from $t = 0$ to $t = 1$. We specify an analytic solution with the functional form

$$f(x, y, t) = \sum_{k,l,m=0}^{3} R^{1,f}_{klm} \sin(xk + yl + \pi tm) + R^{2,f}_{klm} \cos(xk + yl + \pi tm) \quad (8)$$

for the PDF of the process $\mathbf{X}(t)$. The same functional form is used for the coefficients $(\sigma_{ij})^2_{i,j=1}$ of the diffusion matrix σ:

$$\sigma_{ij}(x, y, t) = \sum_{k,l,m=1}^{3} R^{1,\sigma_{ij}}_{klm} \sin(xk + yl + \pi tm) + R^{2,\sigma_{ij}}_{klm} \cos(xk + yl + \pi tm). \quad (9)$$

At the beginning of the simulation, the coefficients $R^{1,f}_{klm}, R^{2,f}_{klm}, R^{1,\sigma_{ij}}_{klm}, R^{2,\sigma_{ij}}_{klm}$ are assigned randomly from a standard normal distribution. Then, all of the coefficients are rescaled and a constant offset is added, in order to enforce

$$\begin{aligned}
&\min_{x,y,t}(f(x, y, t)) = \frac{1}{8\pi^2}, \max_{x,y,t}(f(x, y, t)) = \frac{3}{8\pi^2} \\
&\min_{x,y,t}(\sigma_{ij}(x, y, t)) = 0.6, \max_{x,y,t}(\sigma_{ij}(x, y, t)) = 1.4, \text{ for } i = j \\
&\min_{x,y,t}(\sigma_{ij}(x, y, t)) = -0.3, \max_{x,y,t}(\sigma_{ij}(x, y, t)) = 0.3, \text{ for } i \neq j,
\end{aligned} \quad (10)$$

thus ensuring that $f(x, y, t)$ is positive and integrates to 1, and that $\sigma(x, y, t)$ is positive definite. We then specify $\mathbf{D}(x, y, t)$ such that the Fokker-Planck equation is satisfied. Denoting by $\tilde{f}(x, y, t)$ the PDF of a given numerical approximation $\mathbf{Z}(t)$ to $\mathbf{X}(t)$, and denoting by $\tilde{A}^{1,f}_{kl}, \tilde{A}^{2,f}_{kl}$ the coefficients of the Fourier expansion of $\tilde{f}(x, y, t = 1)$,

$$\tilde{f}(x, y, t = 1) = \sum_{k,l \in (-\infty, +\infty)} \tilde{A}^{1,f}_{kl} \sin(xk + yl) + \tilde{A}^{2,f}_{kl} \cos(xk + yl) \quad (11)$$

with a similar definition for the Fourier coefficients $A^{1,f}_{kl}, A^{2,f}_{kl}$ of $f(x, y, t = 1)$ we estimate the following measure of error between $f(x, y, t = 1)$ and $\tilde{f}(x, y, t = 1)$

$$\epsilon_f = \sqrt{\sum_{k,l=0}^{4} \left(\tilde{A}_{kl}^{1,f} - A_{kl}^{1,f}\right)^2 + \left(\tilde{A}_{kl}^{2,f} - A_{kl}^{2,f}\right)^2} \tag{12}$$

Note that ϵ_f can be estimated only stochastically from the sample PDF of $\tilde{f}(x, y, t = 1)$. We use a sufficient number of samples of $\mathbf{Z}(t = 1)$ to ensure that the 95% confidence interval for ϵ_f has a width smaller than the sample mean for ϵ_f, and employ a jackknife estimator to reduce bias. In addition to estimating the error, we also measure the computational cost of each numerical scheme, in terms of microseconds per sample computation on a single processor. The machine used was a medium-sized cluster of 35 3.0GHz quad-core Xeon processors, and the numerical test cases were implemented in Fortran 90, using the Intel 10.032 compiler.

The results for this test case are given in Table 1, and in graphical form in Figs. 1 and 2. On Fig. 1, which is a log-log plot of error vs. time step, we see that both the DR2 and KP data points fall close to a line of slope two, which confirms the second-order accuracy of both schemes. On Table 1, it can be seen that DR2 has a slightly higher computational cost - it takes about 20% more time for the same time step size - than the KP scheme. This, however, is offset by the lower error produced by the DR2 scheme - only a third of the KP error - to yield a numerical method which is overall computationally more efficient for this test case. This can be seen on Fig. 2, which is a log-log plot of error vs. computational cost. On this plot, the DR2 data points are closer to the lower left corner of the plot, which indicates that a given level of error can be achieved at lower computational cost by the DR2 scheme.

One-dimensional, isotropic test case - comparison between the DR3 and CP schemes

We have chosen to test DR3 (which is defined in Section 7) on a 1D test case due to the prohibitive computational cost associated with the stochastic verification of the convergence properties of an SDE integration scheme with a high order of accuracy. For example, for a third order scheme, halving the time step requires 2^7 times more computational effort in order to obtain a reasonable confidence interval.

The methodology is similar to that of the 2D test case described above. In particular, we set

$$\begin{aligned} f(x,t) &= 1 + 0.3\sin(x - \pi t) \\ B(x,t) &= 1 + 0.25\sin(x + \pi/3), \end{aligned} \tag{13}$$

with $D(x, t)$ such that the Fokker-Planck equation is satisfied. We simulate on the periodic domain $x \in [0, 2\pi)$, from $t = 0$ to $t = 1$, and we define the error as

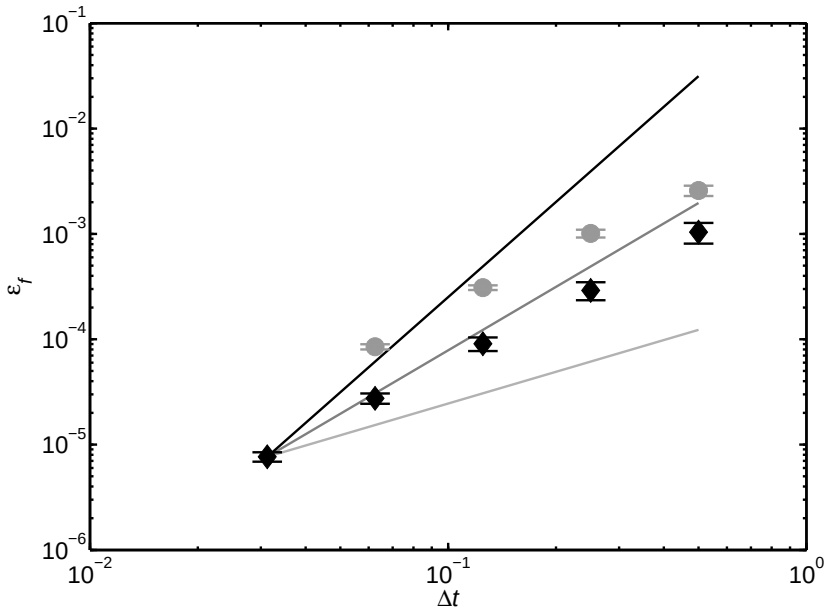

Figure 1: Numerical results comparing the accuracy of the DR2 scheme (black diamonds, horizontal bars denote 95% confidence intervals) with that of the KP scheme (gray circles), for the 2D, anisotropic test case. The light gray, dark gray and black sloped reference lines illustrate respectively first-, second- and third-order convergence.

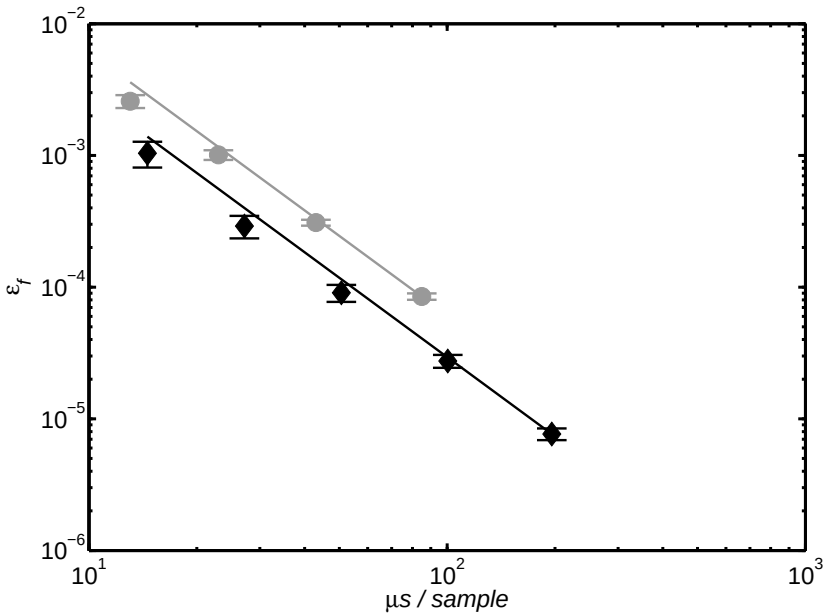

Figure 2: Numerical results comparing the computational efficiency of the DR2 scheme (black diamonds, horizontal bars denote 95% confidence intervals) with that of the KP scheme (gray circles), for the 2D, anisotropic test case. The black sloped reference line illustrates the second-order convergence behavior of DR2. The gray sloped reference line illustrates the second-order convergence behavior of KP.

$$\epsilon_f = \sqrt{\sum_{k=0}^{6} \left(\tilde{A}_k^{1,f} - A_k^{1,f}\right)^2 + \left(\tilde{A}_k^{2,f} - A_k^{2,f}\right)^2}, \tag{14}$$

where again $\tilde{A}_k^{1,f}, \tilde{A}_k^{2,f}$ and $A_k^{1,f}, A_k^{2,f}$ are the Fourier coefficients of $\tilde{f}(x, t = 1)$ and $f(x, t = 1)$ respectively.

The results are presented in Table 2, and in graphical form in Figs. 3 and 4. On Fig. 3, which is a log-log plot of error vs. time step, it can be seen that the CP data points fall close to a line of slope two, confirming the second-order convergence of this scheme, whereas the DR3 scheme achieves third-order convergence for time steps lower than $\Delta t = 1/8$. Also, as can be seen on Table 2, for all time steps the error produced by DR3 is at least 4 times smaller than that produced by CP. On the other hand, the computational cost of DR3 is about 2.5 times higher for a given time step than that of Cao and Pope's scheme. On Fig. 4, which is a log-log plot of error vs. computational cost, it can be seen that at high error levels, CP is more computationally efficient (its data points lie to the left of those of DR3), whereas at low error levels DR3 is more efficient, as it has attained its higher order of convergence.

Based on these numerical test cases, we establish the practical significance of the DR2 and DR3 schemes, which (for fixed Δt) produce less error than other modern SDE integration schemes, have comparable computational efficiency, and can be implemented with ease in a computational code, as we saw from the pseudocode description of DR2.

Finally, we note that as the DRp class of SDE integration schemes are based on an acceptance/rejection procedure on a fixed number of explicit Euler time steps, its stability properties are the same as that of the explicit Euler scheme. While this may prevent DRp from being an appropriate SDE integrator for certain applications, it does not pose a problem in the context of turbulent reactive flow simulations and related applications in mathematical physics, in which the time steps used are small, for the sake of time resolution of the turbulent flow, and for stability of the finite volume solver. Indeed, explicit schemes such as those developed in [12,3] and even explicit Euler are the SDE integrators most often used in this field.

In the subsequent sections, we present the mathematical theory of the general DRp scheme, and prove its properties. We start this with a brief description of Richardson extrapolation and the manner in which it is used by the DRp family of schemes in order to achieve weak p-th order accuracy.

4 Richardson Extrapolation and Its Use by the DRp Scheme

Richardson extrapolation, as introduced in [10], introduces the elegant idea that, if we have a first-order accurate numerical approximation $A^1_{\Delta t}$ to an exact

Case	ϵ_X		ϵ_{dV}	
	RK2	MRK2	RK2	MRK2
4×4, C = 1	1.32×10^{0}	8.17×10^{-1}	4.35×10^{-1}	3.97×10^{-1}
4×4, C = 0.25	4.35×10^{-1}	1.29×10^{-1}	1.07×10^{-1}	4.56×10^{-2}
8×8, C = 1	5.47×10^{-1}	3.80×10^{-1}	1.88×10^{-1}	2.33×10^{-1}
8×8, C = 0.25	7.83×10^{-2}	3.55×10^{-2}	1.84×10^{-2}	1.41×10^{-2}

Table 1: Summary of position and dV errors for Flow 2 at $t = 8.97$.

DR2	Δt	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$
	ϵ_f	1.04e-3	2.91e-4	9.06e-5	2.75e-5	7.67e-6
	95% CI half-width	2.31e-4	5.65e-5	1.34e-5	3.10e-6	7.80e-7
	$\mu s/sample$	14.56	27.16	50.65	100.38	195.92
KP	Δt	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	
	ϵ_f	2.58e-3	1.01e-3	3.09e-4	8.49e-5	
	95% CI half-width	2.90e-4	8.55e-5	1.60e-5	4.80e-6	
	$\mu s/sample$	13.04	22.98	43.01	84.92	

Table 1: Summary of accuracy and computational cost of the DR2 and KP schemes for the 2D anisotropic test case

DR3	Δt	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	
	ϵ_f	2.08e-3	5.62e-4	1.10e-4	1.53e-5	
	95% CI half-width	6.65e-4	8.85e-5	1.05e-5	1.19e-5	
	$\mu s/sample$	7.81	15.60	30.69	60.85	
CP	Δt	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$
	ϵ_f	9.07e-3	2.37e-3	7.32e-4	1.69e-4	4.28e-5
	95% CI half-width	1.40e-3	3.35e-4	8.77e-5	2.13e-5	5.34e-6
	$\mu s/sample$	3.22	6.27	11.82	23.61	47.03

Table 2: Summary of accuracy and computational cost of the DR3 and CP schemes for the 1D isotropic test case

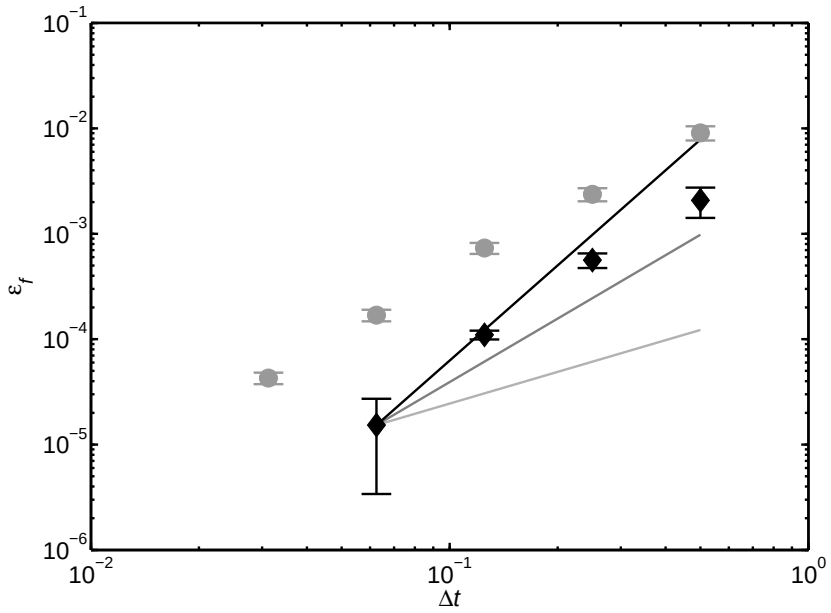

Figure 3: Numerical results comparing the accuracy of the DR3 scheme (black diamonds, horizontal bars denote 95% confidence intervals) with that of the CP scheme (gray circles), for the 1D, isotropic test case. The light gray, dark gray and black sloped reference lines illustrate respectively first-, second- and third-order convergence.

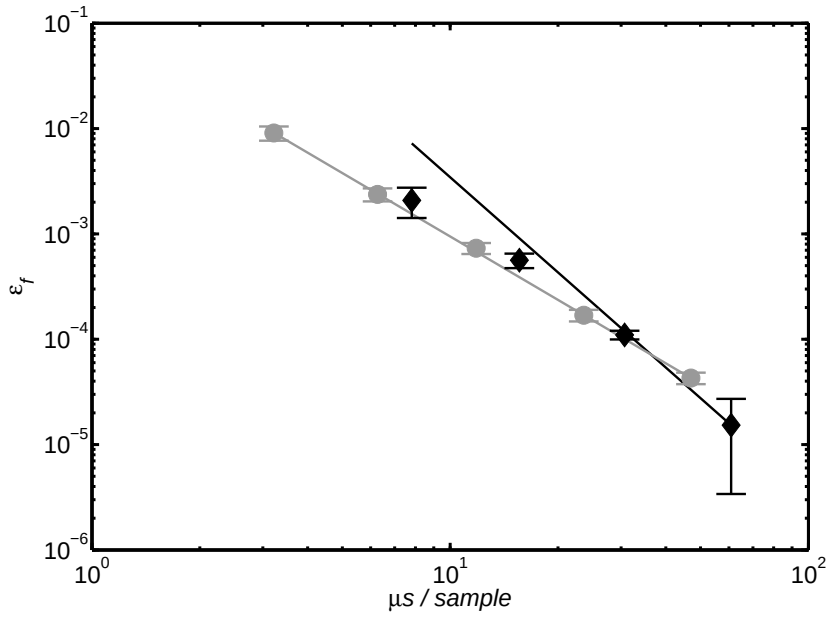

Figure 4: Numerical results comparing the computational efficiency of the DR3 scheme (black diamonds, horizontal bars denote 95% confidence intervals) with that of the CP scheme (gray circles), for the 1D, isotropic test case. The black sloped reference line illustrates the third-order convergence behavior of DR3, for small time steps. The gray sloped reference line illustrates the second-order convergence behavior of CP.

solution A, and the error with respect to some linear functional $g(\cdot)$ varies smoothly:

$$g(A^1_{\Delta t} - A) = \sum_{i=1}^{\infty} K_i \Delta t^i, \tag{15}$$

then we can construct from $A^1_{\Delta t}$ a second-order accurate approximation $A^2_{\Delta t}$ to A by setting $A^2_{\Delta t} = 2A^1_{\Delta t/2} - A^1_{\Delta t}$. Then eq.(15) gives us that

$$\begin{aligned} g(A^2_{\Delta t} - A) &= 2g(A^1_{\Delta t/2} - A) - g(A^1_{\Delta t} - A) \\ &= \sum_{i=1}^{\infty} K_i (2(\Delta t/2)^i - \Delta t^i) \\ &= \sum_{i=2}^{\infty} K_i (2(\Delta t/2)^i - \Delta t^i), \end{aligned} \tag{16}$$

and so we see that the first-order component of the error has vanished. Applying this method inductively, we can obtain a scheme with an arbitrarily high order of accuracy from the first order scheme $A^1_{\Delta t}$ by the following linear combination

$$A^p_{\Delta t} = \sum_{k=1}^{p} l^p_k A^1_{\frac{\Delta t}{2^{k-1}}}, \tag{17}$$

where the coefficients l^p_k satisfy the following recursive relation

$$\begin{aligned} \left(l^2_1, l^2_2\right) &= (-1, 2) \\ \left(l^p_1, ..., l^p_p\right) &= \frac{\left[2^{p-1}\left(0, l^{p-1}_1, ..., l^{p-1}_{p-1}\right) - \left(l^{p-1}_1, ..., l^{p-1}_{p-1}, 0\right)\right]}{2^{p-1} - 1}. \end{aligned} \tag{18}$$

In the context of solutions to stochastic differential equations([4],[9]), Richardson extrapolation has been used to obtain p-th order accurate estimates for the expected value of a function of the SDE solution at the end time, $E\left(g\left(\mathbf{X}(T)\right)\right)$. This can done by computing, for each time step $\Delta t, \Delta t/2, ...$, Euler solutions with that time step, which we shall denote by $\mathbf{Z}^E_{\Delta t}(T), \mathbf{Z}^E_{\Delta t/2}(T), ...$, and approximating $E\left(g\left(\mathbf{X}(T)\right)\right)$ by the expected value of the Richardson extrapolate of $g\left(\mathbf{Z}^E_{\Delta t}(T)\right)$:

$$E\left(g\left(\mathbf{X}(T)\right)\right) \approx E\left(\sum_{k=1}^{p} l^p_k g\left(\mathbf{Z}^E_{\frac{\Delta t}{2^{k-1}}}(T)\right)\right). \tag{19}$$

The approximation given by eq.(19), while elegant and effective, is applicable only if we are interested in a functional of the solution at the end time, T. On the other hand, in many applications it is necessary to use an SDE integration procedure which gives an accurate result at each intermediate time step, due

to the fact that the SDE is coupled to another process. As an example, in the implementation of a Monte Carlo method for turbulent combustion [13], an overall time step may consist of a transport substep (in which an SDE of the form of eq.(1) is solved), followed by a reaction substep and a diffusion substep. As the last of these substeps uses the values provided by the first, it is easily seen that the transport substep needs to employ an SDE integration scheme which is accurate at intermediate times as well.

With this in mind, we adopt an alternative way of performing Richardson extrapolation on the Euler SDE solutions. Without loss of generality, let $\mathbf{X}(t=0)=\mathbf{0}$, let, for $k=0,...,p-1$, $f^k_{\Delta t}(\mathbf{x})$ be the probability density functions (PDFs) of the random variables $\mathbf{Z}^E_{\frac{\Delta t}{2^k}}(\Delta t)$, respectively, and let $f_{\mathbf{X}(\Delta t)}(\mathbf{x})$ be the PDF of the random variable $\mathbf{X}(\Delta t)$ which we are approximating numerically. Furthermore, let us denote by $\mathbf{Z}^{DRp}_{\Delta t}$ the random variable which is the DRp solution after a single time step of length Δt, and let $f^{DRp}_{\Delta t}(\mathbf{x})$ be its PDF.

Following [12], we note that a sufficient condition for the weak p-th order accuracy of the DRp scheme is that it satisfies, for any multi-index of non-negative integers $(i_1,i_2,...,i_n)$ with $\sum_{m=1}^n i_m \leq 2p+2$, the inequality

$$\left| E\left(\prod_{m=1}^{n} \left(Z^{DRp}_{\Delta t,m} \right)^{i_m} - \prod_{m=1}^{n} \left(X_m(\Delta t) \right)^{i_m} \right) \right| \leq \tilde{C} \Delta t^{p+1}, \tag{20}$$

where $\tilde{C}$ is a positive constant which only depends on n,p and the fields $\mathbf{D}(\mathbf{x},t),\sigma(\mathbf{x},t)$. Previously ([9],[11]) it has also been demonstrated that for fields $\mathbf{D}(\mathbf{x},t),\sigma(\mathbf{x},t)$ which are sufficiently smooth, there exist constants $C^E_1, C^E_2, ..., C^E_{p+1}$ such that

$$\left| E\left(\prod_{m=1}^{n} \left(Z^E_{\Delta t,m} \right)^{i_m} - \prod_{m=1}^{n} \left(X_m(\Delta t) \right)^{i_m} \right) - \sum_{m=1}^{p} C^E_m \Delta t^m \right| \leq \tilde{C}^E_{p+1} \Delta t^{p+1}, \tag{21}$$

and hence

$$\left| E\left(\sum_{k=1}^{p} l^p_k \prod_{m=1}^{n} \left(Z^E_{\frac{\Delta t}{2^{k-1}},m} \right)^{i_m} - \prod_{m=1}^{n} \left(X_m(\Delta t) \right)^{i_m} \right) \right| \leq \tilde{C}^E_{p+1} \Delta t^{p+1}, \tag{22}$$

We have designed the DRp solution so that $f^{DRp}_{\Delta t}(\mathbf{x})$ satisfies

$$\int \left| f^{DRp}_{\Delta t}(\mathbf{x}) - \sum_{k=0}^{p-1} l^p_{k+1} f^k_{\Delta t}(\mathbf{x}) \right| |\mathrm{d}\mathbf{x}| \leq C \Delta t^{p+1}, \tag{23}$$

for a given finite constant C, independent of Δt. It is proved in Theorem 2 in the Appendix that eq.(23) implies that

$$\left| E\left(\prod_{m=1}^{n} \left(Z^{DRp}_{\Delta t,m} \right)^{i_m} - \sum_{k=1}^{p} l^p_k \prod_{m=1}^{n} \left(Z^E_{\frac{\Delta t}{2^{k-1}},m} \right)^{i_m} \right) \right| \leq C' \Delta t^{p+1}, \tag{24}$$

where C' is another constant which only depends on n, p and the fields $\mathbf{D}(\mathbf{x}, t), \sigma(\mathbf{x}, t)$. Since eq.(20) follows directly from eqs.(22,24), we see that eq.(23) and Theorem 2 imply that the DRp scheme is weak p-th order accurate.

In the remainder of this paper, we give a general description of the DRp schemes and a proof of eq.(23). First, however, we need to specify smoothness and boundedness criteria on the SDE drift and diffusion fields, which are necessary for the correct operation of the DRp schemes.

5 Smoothness Requirements of the DRp Scheme

As previously mentioned, we are computing, in $\mathbb{R}^n \times [0, T]$, weak solutions to the SDE problem as given by eq.(1). We require that the fields $\mathbf{D}, \sigma$ be smooth (all derivatives exist and are bounded), so that the result of eq.(21), derived in [11], holds true. Furthermore, for the correct operation of the DRp family of numerical schemes, we require that the drift vector field, $\mathbf{D}(\mathbf{x}, t) \in R^n$, be bounded:

$$\|\mathbf{D}(\mathbf{x}, t)\| \leq C_1^D, \tag{25}$$

and globally Lipschitz continuous in both space and time:

$$\|\mathbf{D}(\mathbf{x_1}, t_1) - \mathbf{D}(\mathbf{x_2}, t_2)\| \leq C_2^{D,\mathbf{x}} \|\mathbf{x}_1 - \mathbf{x}_2\| + C_2^{D,t} |t_1 - t_2|, \tag{26}$$

and that the diffusion field of matrices, $\sigma(\mathbf{x}, t) \in \mathrm{Mat}(n, n)$, be globally bounded:

$$\|\sigma(\mathbf{x}, t)\mathbf{v}\| \leq C_1^\sigma \|\mathbf{v}\|, \text{ for any } \mathbf{v} \in R^n, \tag{27}$$

as well as being globally coercive:

$$\|\sigma(\mathbf{x}, t)\mathbf{v}\| \geq C_2^\sigma \|\mathbf{v}\|, \text{ for any } \mathbf{v} \in R^n, \tag{28}$$

and globally Lipschitz continuous with respect to the matrix norm:

$$\|\sigma(\mathbf{x_1}, t_1) - \sigma(\mathbf{x_2}, t_2)\| \leq C_3^{\sigma,\mathbf{x}} \|\mathbf{x}_1 - \mathbf{x}_2\| + C_3^{\sigma,t} |t_1 - t_2|, \tag{29}$$

Here it is understood that constants $C_1^D, C_2^{D,\mathbf{x}}, C_2^{D,t}, C_1^\sigma, C_2^\sigma, C_3^{\sigma,\mathbf{x}}, C_3^{\sigma,t}$ are finite and strictly positive. Note also that eqs.(27 - 29) imply similar regularity conditions on σ^{-1}, the matrix inverse of σ:

$$\left\|\sigma^{-1}(\mathbf{x}, t)\mathbf{v}\right\| \leq C_1^{\sigma^{-1}} \|\mathbf{v}\|, \text{ for any } \mathbf{v} \in R^n, \tag{30}$$

$$\left\|\sigma^{-1}(\mathbf{x}, t)\mathbf{v}\right\| \geq C_2^{\sigma^{-1}} \|\mathbf{v}\|, \text{ for any } \mathbf{v} \in R^n, \tag{31}$$

and

$$\|\sigma^{-1}(\mathbf{x_1}, t_1) - \sigma^{-1}(\mathbf{x_2}, t_2)\| \leq C_3^{\sigma^{-1},\mathbf{x}} \|\mathbf{x}_1 - \mathbf{x}_2\| + C_3^{\sigma^{-1},t} |t_1 - t_2|, \tag{32}$$

where $C_1^{\sigma^{-1}} = \frac{1}{C_2^\sigma}$, $C_2^{\sigma^{-1}} = \frac{1}{C_1^\sigma}$, $C_3^{\sigma^{-1},\mathbf{x}} = \frac{C_3^{\sigma,\mathbf{x}}}{\left(C_2^\sigma\right)^2}$ and $C_3^{\sigma^{-1},t} = \frac{C_3^{\sigma,t}}{\left(C_2^\sigma\right)^2}$ are again finite and strictly positive constants. For the sake of compactness of notation, we shall call $C_1^D, C_2^{D,\mathbf{x}}, C_2^{D,t}, C_1^\sigma, C_2^\sigma, C_3^{\sigma,\mathbf{x}}, C_3^{\sigma,t}, C_1^{\sigma^{-1}}, C_2^{\sigma^{-1}}, C_3^{\sigma^{-1},\mathbf{x}}, C_3^{\sigma^{-1},t}$ the Richardson regularity constants, and we shall denote them collectively as $\{C\}$.

We note that in the application of Lagrangian Monte Carlo turbulent reactive flow simulations, the above regularity conditions hold for simulations on well-resolved finite volume grids and in the absence of compressible shocks (i.e., a subsonic simulation).

6 Framework of the DRp Scheme

Here we describe the random variable $\mathbf{Z}_{\Delta t}^{DRP}$, and demonstrate why its PDF satisfies eq.(23). Due to the fact that the Richardson extrapolation vectors l_k^p have negative components, eq.(23) requires that the PDF of $\mathbf{Z}_{\Delta t}^{DRP}$ approximate a non-convex linear combination of other PDFs, which are in themselves easily sampled from. To achieve this goal, we use an acceptance-rejection approach, with $f_{\Delta t}^{p-1}(\mathbf{x})$ as the instrumental distribution.

An additional concept which we need to achieve this is that of the 2^{p-1}-step sample path which corresponds to the random variables $\mathbf{Z}_{\frac{\Delta t}{2^k}}^E$. This concept embodies the idea that if we do not update the coefficients for the second step, two Euler steps of length $\Delta t/2$ produce exactly the same result as a single Euler step of length Δt, and hence we can use time steps of length $\frac{\Delta t}{2^{p-1}}$ to sample from each of $\mathbf{Z}_{\frac{\Delta t}{2^k}}^E$, by updating the values of $\mathbf{D}(\mathbf{x},t), \sigma(\mathbf{x},t)$ only when we reach a time which is an integer multiple of $\frac{\Delta t}{2^k}$, instead of at each time step.

More concretely, let us denote $\lfloor i \rfloor_{p,k} \equiv \left\lfloor \frac{i}{2^{p-1-k}} \right\rfloor 2^{p-1-k}$, let $\eta_i \sim \mathcal{N}(\mathbf{0}, I)$ be independent samples from the standard normal distribution, and define the random variable $\mathbf{U}^{(k)}$, for each $k = 0, 1, ..., p-1$ as $\mathbf{U}^{(k)} = \left(\mathbf{U}_i^{(k)}\right)_{i=1}^{2^{p-1}}$, where $\mathbf{U}_i^{(k)}$ are defined by:

$$\begin{aligned}
\mathbf{U}_0^{(k)} &= \mathbf{0} \\
\mathbf{U}_i^{(k)} &= \mathbf{U}_{i-1}^{(k)} + \mathbf{D}\left(\mathbf{U}_{\lfloor i-1 \rfloor_{p,k}}^{(k)}, \lfloor i-1 \rfloor_{p,k} \frac{\Delta t}{2^{p-1}}\right) \frac{\Delta t}{2^{p-1}} \\
&+ \sigma\left(\mathbf{U}_{\lfloor i-1 \rfloor_{p,k}}^{(k)}, \lfloor i-1 \rfloor_{p,k} \frac{\Delta t}{2^{p-1}}\right) \eta_i \sqrt{\frac{\Delta t}{2^{p-1}}}. \qquad (33)
\end{aligned}$$

Finally, let $f_{\Delta t}^{U,k}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})$ be the probability density function of $\mathbf{U}^{(k)}$, i.e. the PDF of the event $\left\{\mathbf{U}_i^{(k)} = \mathbf{x}_i\right\}$. This variable, $\mathbf{U}^{(k)}$ is what we will call the 2^{p-1}-step sample path generated by the 2^k-step Euler scheme. Note that it is an $n2^{p-1}$-dimensional random variable, and correspondingly, its PDF is defined on an $n2^{p-1}$-dimensional sample space.

Noting that in the definition of $\mathbf{U}^{(k)}$, the coefficients $\mathbf{D}, \sigma$ are updated only when the number of the time step is an integer multiple of 2^{p-1-k}, we have that

$$\begin{aligned}\mathbf{U}^{(k)}_{i2^{p-1-k}} &= \mathbf{U}^{(k)}_{(i-1)2^{p-1-k}} + \mathbf{D}\left(\mathbf{U}^{(k)}_{(i-1)2^{p-1-k}}, (i-1)\frac{\Delta t}{2^k}\right)\frac{\Delta t}{2^k} \\ &\quad + \sigma\left(\mathbf{U}^{(k)}_{(i-1)2^{p-1-k}}, (i-1)\frac{\Delta t}{2^k}\right)\left[\sum\nolimits_{j=(i-1)2^{p-1-k}+1}^{i2^{p-1-k}} \eta_j \sqrt{\frac{\Delta t}{2^{p-1}}}\right] \quad (34)\end{aligned}$$

and so, since the vectors η_j are independent standard normal random variables, we have that the term in the square brackets in eq.(34) is distributed as $N\left(0, \frac{\Delta t}{2^k}\right)$, which implies that the random variables $\left\{\mathbf{U}^{(k)}_{i2^{p-1-k}} | \mathbf{U}^{(k)}_{(i-1)2^{p-1-k}} = \mathbf{x}\right\}$ and $\left\{\mathbf{Z}^k_i | \mathbf{Z}^k_{i-1} = \mathbf{x}\right\}$ are identically distributed, in other words the 2^{p-1-k} steps corresponding to eq.(34) are identical to a singe Euler step of length $\frac{\Delta t}{2^k}$.

The above paragraph implies that, for each $k = 0, 1, ..., p-1$, the PDF $f^k_{\Delta t}(\mathbf{x})$ is the marginal PDF of $f^{U,k}_{\Delta t}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})$ in the last variable, and so we can achieve the goal of eq.(23) by approximating a linear combination of the PDFs of the 2^{p-1}-step sample paths.

The reason for this approach is the rather unexpected result that, if we compute a realization of $\mathbf{U}^{(p-1)}$, we can also compute exactly the ratio $\frac{f^{U,k}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)}{f^{U,p-1}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)}$. To see how, define $f^{U,k}_{\Delta t,U_j}(\mathbf{x}_j; \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{j-1})$ to be the PDF of $\mathbf{U}^{(k)}_j$, conditional upon $\left\{\mathbf{U}^{(k)}_1 = \mathbf{x}_1, \mathbf{U}^{(k)}_2 = \mathbf{x}_2, ..., \mathbf{U}^{(k)}_{j-1} = \mathbf{x}_{j-1}\right\}$, and note that

$$f^{U,k}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right) = \prod_{j=1}^{2^{p-1}} f^{U,k}_{\Delta t,U_j}\left(\mathbf{U}^{(p-1)}_j; \mathbf{U}^{(p-1)}_1, \mathbf{U}^{(p-1)}_2, ..., \mathbf{U}^{(p-1)}_{j-1}\right), \quad (35)$$

which gives us that

$$\frac{f^{U,k}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)}{f^{U,p-1}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)} = \prod_{j=1}^{2^{p-1}} \frac{f^{U,k}_{\Delta t,U_j}\left(\mathbf{U}^{(p-1)}_j; \mathbf{U}^{(p-1)}_1, \mathbf{U}^{(p-1)}_2, ..., \mathbf{U}^{(p-1)}_{j-1}\right)}{f^{U,p-1}_{\Delta t,U_j}\left(\mathbf{U}^{(p-1)}_j; \mathbf{U}^{(p-1)}_1, \mathbf{U}^{(p-1)}_2, ..., \mathbf{U}^{(p-1)}_{j-1}\right)}, \quad (36)$$

where the factors of the above expression are easily evaluated, as they are just evaluations of the joint normal distributions which correspond to the Euler steps in eq.(33):

$$\begin{aligned}&f^{U,k}_{\Delta t,U_j}(\mathbf{x}_j; \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{j-1}) = \\ &\quad \frac{1}{(2\pi)^{n/2}\left|B(\mathbf{x}^*, t^*)\frac{\Delta t}{2^{p-1}}\right|^{1/2}} \exp\left(-\mathbf{v}^T \frac{1}{2}\left(B(\mathbf{x}^*, t^*)\frac{\Delta t}{2^{p-1}}\right)^{-1}\mathbf{v}\right), \quad (37)\end{aligned}$$

where, for the sake of brevity in the above equation, we use the notation

$$\mathbf{x}^* = \mathbf{x}_{\lfloor j-1 \rfloor_{p,k}}; t^* = \lfloor j-1 \rfloor_{p,k} \frac{\Delta t}{2^{p-1}} \tag{38}$$

and

$$\mathbf{v} = \left[\mathbf{x}_j - \mathbf{x}_{j-1} - \mathbf{D}\left(\mathbf{x}^*, t^*\right) \frac{\Delta t}{2^{p-1}} \right]. \tag{39}$$

Note that eqs.(36 - 39) require values of $\mathbf{D}(\mathbf{x}), \sigma(\mathbf{x})$ at $\mathbf{x} = \mathbf{U}^* = \mathbf{U}^{(p-1)}_{\lfloor j-1 \rfloor_{p,k}}$ only, and in the process of computing a realization of $\mathbf{U}^{(p-1)}$ (i.e., taking 2^{p-1} Euler steps of length $\frac{\Delta t}{2^{p-1}}$) we have already computed these values, so no further sampling of the diffusion and drift fields is needed in order to evaluate the products in eq.(36).

And so, we are ready to proceed to the pseudocode description of the DRP scheme.

7 Pseudocode Description of the DRp Scheme

Here, we give a pseudocode description of the p-th order accurate Direct Richardson Scheme. As in the previous sections we assume, without loss of generality, that the initial location is $\mathbf{Z}(t=0) = \mathbf{0}$, and we describe the algorithm by which we calculate $\mathbf{Z}^{DRp}_{\Delta t}$. First, we choose a parameter $c \in (0, 0.5)$ which will serve as a lower bound for the acceptance probability - in the computational results presented in this work, the value $c = 0.1$ is used. Later on, we shall demonstrate that the unconditional acceptance probability of the DRp algorithm converges to 0.5 in the limit $\Delta t \downarrow 0$.

1. Obtain a sample of the random variable $\mathbf{U}^{(p-1)}$, according to eq.(33) (i.e., take 2^{p-1} Euler time steps).

2. For each $k = 0, 1, ..., p-2$, evaluate $\frac{f^{U,k}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)}{f^{U,p-1}_{\Delta t}\left(\mathbf{U}^{(p-1)}\right)}$ according to eqs.(36 - 39), and calculate

$$H\left(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}\right) \equiv \frac{1}{2} \sum_{k=0}^{p-1} l^p_{k+1} \frac{f^{U,k}_{\Delta t}\left(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}\right)}{f^{U,p-1}_{\Delta t}\left(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}\right)}. \tag{40}$$

   at $\left(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}\right) = \mathbf{U}^{(p-1)}$.

3. Sample a random variable ξ with a standard uniform distribution: $\xi \sim U(0,1)$

4. If the following acceptance criterion is met

$$\xi < \max\left(H\left(\mathbf{U}^{(p-1)}\right), c\right), \tag{41}$$

   then set $\mathbf{Z}^{DRp}_{\Delta t} = \mathbf{U}^{(p-1)}_{2^{p-1}}$. Else, go back to step 1.

As can be seen, the DR2 algorithm described earlier is the particular case of the above algorithm when $p = 2$ and $c = 0.1$. Examining eq.(41) it is easily seen that the probability of acceptance at each iteration of steps 1-4 is at least equal to c. In fact, in the Appendix we shall prove the following theorem, which implies that as the time step Δt decreases, the probability of acceptance converges to $\frac{1}{2}$:

Theorem 1: For any integer $m, p \geq 1$, and any real number $d > 0$, if the fields $\mathbf{D}(\mathbf{x}, t)$ and $\sigma(\mathbf{x}, t)$ satisfy the regularity conditions stated in eqs.(25 - 32), then there exists a constant $\bar{C} \in (0, \infty)$, dependent only on $p, m, d, \{C\}$, such that if the set $E \subset R^{n2^{p-1}}$ is defined by

$$E = \left\{ (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) \in R^{n2^{p-1}} | \left| \frac{f_{\Delta t}^{U,k}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})}{f_{\Delta t}^{U,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})} - 1 \right| > d \right\}, \tag{42}$$

then

$$\int_E f_{\Delta t}^{U,k}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) |d\mathbf{x}_1| |d\mathbf{x}_2| ... |d\mathbf{x}_{2^{p-1}}| \leq \bar{C} \Delta t^m \tag{43}$$

and

$$\int_E f_{\Delta t}^{U,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) |d\mathbf{x}_1| |d\mathbf{x}_2| ... |d\mathbf{x}_{2^{p-1}}| \leq \bar{C} \Delta t^m. \tag{44}$$

Theorem 1 as a very powerful result, as it implies that the probability of the quantity $H\left(\mathbf{U}^{(p-1)}\right)$ differing considerably from $\frac{1}{2}$ on a given acceptance-rejection step decreases faster than any power of Δt, as $\Delta t \downarrow 0$. This implies that the unconditional acceptance probability of the DRp algorithm converges to $\frac{1}{2}$, as $\Delta t \downarrow 0$.

To see why this is so, note that by the definition of the Richardson extrapolation coefficients, eq.(18), we have that $\sum_{k=0}^{p-1} l_{k+1}^p = 1$, hence $H\left(\mathbf{U}^{(p-1)}\right)$ differing considerably from $\frac{1}{2}$ implies that at for least one $k \in \{0, 1, 2, ..., p-1\}$, $\frac{f_{\Delta t}^{U,k}\left(\mathbf{U}^{(p-1)}\right)}{f_{\Delta t}^{U,p-1}\left(\mathbf{U}^{(p-1)}\right)}$ differs considerably from 1, and by Theorem 1 the probability of this is negligible.

More concretely, choosing an arbitrary $c' \in \left(c, \frac{1}{2}\right)$ and setting $d = \frac{1-2c'}{\sum_{k=0}^{p-1} |l_{k+1}^p|}$, we get that

$$H\left(\mathbf{U}^{(p-1)}\right) \notin [c', 1-c'] \tag{45}$$

implies that for at least one $k \in \{0, 1, ..., p-1\}$ we have that $\left| \frac{f_{\Delta t}^{U,k}\left(\mathbf{U}^{(p-1)}\right)}{f_{\Delta t}^{U,p-1}\left(\mathbf{U}^{(p-1)}\right)} - 1 \right| > d$. However, denoting by $P_{\mathbf{U}^{(p-1)}}\{\cdot\}$ the probability of a given event (dependent

on $\mathbf{U}^{(p-1)}$) for one sampling of $\mathbf{U}^{(p-1)}$, i.e., one acceptance-rejection step, we have that

$$P_{\mathbf{U^{(p-1)}}}\left\{\left|\frac{f_{\Delta t}^{U,k}\left(\mathbf{U}^{(p-1)}\right)}{f_{\Delta t}^{U,p-1}\left(\mathbf{U}^{(p-1)}\right)}-1\right|>d\right\}=\int_{E}f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}}\right)\left|\mathrm{d}\mathbf{x}_1\right|...\left|\mathrm{d}\mathbf{x}_{2^{p-1}}\right|, \tag{46}$$

where E is as defined in eq.(42). Hence, applying Theorem 1 for this value of d, an arbitrary $m \geq p+1$ and all $k = 0, 1, ..., p-1$, we get that there exists a constant $\bar{C}$ such that

$$\begin{aligned} &P_{\mathbf{U^{(p-1)}}}\left\{H\left(\mathbf{U}^{(p-1)}\right)\notin[c',1-c']\right\}\leq \\ &\leq\textstyle\sum_{k=0}^{p-1}P_{\mathbf{U^{(p-1)}}}\left\{\left|\frac{f_{\Delta t}^{U,k}\left(\mathbf{U}^{(p-1)}\right)}{f_{\Delta t}^{U,p-1}\left(\mathbf{U}^{(p-1)}\right)}-1\right|>d\right\}\leq\bar{C}\Delta t^m \end{aligned} \tag{47}$$

Since $c' \in \left(c, \frac{1}{2}\right)$ and $m \geq p+1$ are arbitrary, eq.(47) implies that for an arbitrarily narrow interval $[c', 1-c']$, centered on $\frac{1}{2}$, as $\Delta t \downarrow 0$, the probability that $H\left(\mathbf{U}^{(p-1)}\right)$ falls outside of this interval decreases faster than any power of Δt. This, combined with eq.(41) (which implies that the probability of acceptance is at least c, on each acceptance-rejection step of the DRp algorithm), implies that the DRp acceptance-rejeciton algorithm is well-behaved, and as $\Delta t \downarrow 0$, the expected number of steps to acceptance converges to 2.

In the next section, we use the result of Theorem 1 to prove that, with $\mathbf{Z}_{\Delta t}^{DRp}$ being generated by the above algorithm, $f_{\Delta t}^{DRp}$ satisfies eq.(23).

8 Proving that the DRp Scheme Satisfies Its Objective

Before we proceed, we need some additional definitions. Let us define $V_{c'}$ as follows:

$$V_{c'}=\left\{\left(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}}\right)\in R^{n2^{p-1}}\left|H\left(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}}\right)\in[c',1-c']\right.\right\} \tag{48}$$

It is important to note that, by Theorem 1, the complement of $V_{c'}$, $\left(V_{c'}\right)^c$ contains a negligible part of the mass of the PDFs $f_{\Delta t}^{U,k}\left(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}}\right)$, in the sense that

$$\int_{\left(V_{c'}\right)^c}f_{\Delta t}^{U,k}\left(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}}\right)\left|\mathrm{d}\mathbf{x}_1\right|...\left|\mathrm{d}\mathbf{x}_{2^{p-1}}\right|\leq\bar{C}\Delta t^m. \tag{49}$$

Next, let $\mathbf{W}^{(p-1)}=\left(\mathbf{W}_i^{(p-1)}\right)_{i=1}^{2^{p-1}}$ be the value of $\mathbf{U}^{(p-1)}$ at the last step of the acceptance-rejection loop, i.e. the value which leads to an acceptance, and let $f_{\Delta t}^{W,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ be its PDF on the sample space $R^{n2^{p-1}}$. Finally,

let $P_{\mathbf{W}^{(\mathbf{p}-\mathbf{1})}}\{\cdot\}$ denote the probability of a given event dependent on $\mathbf{W}^{(p-1)}$. Note that the difference between $P_{\mathbf{U}^{(\mathbf{p}-\mathbf{1})}}\{\cdot\}$ and $P_{\mathbf{W}^{(\mathbf{p}-\mathbf{1})}}\{\cdot\}$ is that the former is the probability for one acceptance-rejection step, whereas the latter is the probability for the entire time step, i.e., until the algorithm results in acceptance.

We have then that

$$\mathbf{Z}_{\Delta t}^{DRp} = \mathbf{W}_{2^{p-1}}^{(p-1)} \tag{50}$$

and

$$f_{\Delta t}^{DRp}(\mathbf{x}) = \int_{R^{n(2^{p-1}-1)}} f_{\Delta t}^{W,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}-1}, \mathbf{x}) |\mathrm{d}\mathbf{x}_1| |\mathrm{d}\mathbf{x}_2| ... |\mathrm{d}\mathbf{x}_{2^{p-1}-1}|. \tag{51}$$

Now, by eq.(48), we have that, on $V_{c'}$, $\max(H(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}), c) = H(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \in [c', 1-c']$, and so for $(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \in V_{c'}$

$$\begin{gathered}\max(H(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}), c) \times f_{\Delta t}^{U,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) = \\ \frac{1}{2}\sum_{k=0}^{p-1} l_{k+1}^p \frac{f_{\Delta t}^{U,k}(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}})}{f_{\Delta t}^{U,p-1}(\mathbf{x}_1,...,\mathbf{x}_{2^{p-1}})} \times f_{\Delta t}^{U,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) = \\ \frac{1}{2}\sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}})\end{gathered} \tag{52}$$

Therefore, since $f_{\Delta t}^{U,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}})$ is the instrumental distribution in the DRp acceptance-rejection scheme, and the acceptance probability is as defined in eq.(41), we have that, on $V_{c'}$:

$$\frac{f_{\Delta t}^{W,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})}{\sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}})} = Q_1, \tag{53}$$

where Q_1 is a constant that does not vary with $(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})$, and hence, because $f_{\Delta t}^{W,p-1}$ is strictly positive, we have that

$$\begin{gathered}\int_{V_{c'}} \left| f_{\Delta t}^{W,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) - \sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \right| |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| = \\ \left| \int_{V_{c'}} f_{\Delta t}^{W,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) - \sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| \right| \leq \\ \int_{(V_{c'})^c} f_{\Delta t}^{W,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| \\ + \int_{(V_{c'})^c} \left| \sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \right| |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}|,\end{gathered} \tag{54}$$

where, in order to get the inequality in the above eq.(54), we used the triangle inequality and the fact that $f_{\Delta t}^{W,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}})$ and $\sum_{k=0}^{p-1} l_{k+1}^p f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}})$ both integrate to 1 over the entire space $R^{n2^{p-1}}$. The above result, and eq.(51), imply that

$$\int \left| f_{\Delta t}^{DRP}(\mathbf{x}) - \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{k}(\mathbf{x}) \right| |\mathrm{d}\mathbf{x}| =$$
$$\int_{R^{n2^{p-1}}} \left| f_{\Delta t}^{W,p-1}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) - \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \right| |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| \leq$$
$$2 \int_{(V_{c'})^c} f_{\Delta t}^{W,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}|$$
$$+2 \int_{(V_{c'})^c} \left| \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \right| |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}|, \tag{55}$$

Now, by eq.(49), we have that

$$\int_{(V_{c'})^c} \left| \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{U,k}(\mathbf{x}_1, ..., \mathbf{x}_{2^{p-1}}) \right| |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| \leq \bar{C} \Delta t^m \sum_{k=0}^{p-1} \left| l_{k+1}^{p} \right|, \tag{56}$$

and by the definition of $f_{\Delta t}^{W,p-1}$ we have that

$$\int_{(V_{c'})^c} f_{\Delta t}^{W,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) |\mathrm{d}\mathbf{x}_1| ... |\mathrm{d}\mathbf{x}_{2^{p-1}}| = P_{\mathbf{W}^{(p-1)}} \left\{ \mathbf{W}^{(p-1)} \in (V_{c'})^c \right\} \tag{57}$$

Also, by eq.(47) and the fact that the acceptance probability is at least c, we have that

$$P_{\mathbf{W}^{(p-1)}} \left\{ \mathbf{W}^{(p-1)} \in (V_{c'})^c \right\} \leq$$
$$\frac{1}{c} P_{\mathbf{U}^{(p-1)}} \left\{ \mathbf{U}^{(p-1)} \in (V_{c'})^c \right\} \leq \frac{1}{c} \bar{C} \Delta t^m \tag{58}$$

Substituting the result of eqs.(56,57,58) into eq.(55), we get that

$$\int \left| f_{\Delta t}^{DRp}(\mathbf{x}) - \sum_{k=0}^{p-1} l_k^p f_{\Delta t}^{k}(\mathbf{x}) \right| |\mathrm{d}\mathbf{x}| \leq 2 \bar{C} \Delta t^m \left(\frac{1}{c} + \sum_{k=0}^{p-1} \left| l_{k+1}^{p} \right| \right), \tag{59}$$

which proves that the DRp scheme satisfies the required condition, eq.(23). In fact, as $m \geq p+1$ can be arbitrarily large, the result of eq.(59) is much stronger than the required condition of eq.(23): we have just shown that the L^1 difference between $f_{\Delta t}^{DRp}(\mathbf{x})$ and the Richardson extrapolate, $\sum_{k=0}^{p-1} l_k^p f_{\Delta t}^{k}(\mathbf{x})$, decreases faster than any power of Δt, as $\Delta t \downarrow 0$. This concludes the proof that the DRp scheme is weak p-th order accurate.

9 Summary and Conclusions

We have developed a new class of weak p-th order accurate SDE integration schemes, for the solution of non-homogeneous, anisotropic Ito SDE with strictly positive definite diffusion matrices. These schemes, called the Direct Richardson

p-th order accurate (DRp) schemes, perform Richardson extrapolation on the Euler algorithm in a conceptually new way, by means of an acceptance-rejection algorithm, after each time step.

Unlike previous applications of Richardson extrapolation to an Euler SDE solution, which are only applicable to the problem of estimating functionals of the distribution of the SDE process at the end time, the DRp solution is weak p-th order accurate at each time step of the simulation, and can therefore be applied to any problem which requires a weakly convergent SDE integration scheme.

A simplified description of a particular member of the DRp class, DR2, has been provided. This description illustrates the elegance of the Direct Richardson schemes, and their ease of implementation in a computational code. Numerical results have been provided for both 2D anisotropic and 1D isotropic test cases, which compare the performance of DR2 and DR3 with that of other modern SDE integration schemes, in particular those developed by Kloeden and Platen [4] and by Cao and Pope [3]. The numerical results indicate that the error of the DR2, DR3 schemes is smaller than that of existing schemes based on Ito-Taylor expansions, whereas the computational cost of the DR schemes is somewhat higher, so that the overall computational efficiency is comparable. This suggests that the DRp family of SDE integration schemes are a practicable alternative to existing SDE integration schemes, with the benefit of being easier to implement, especially in cases where the SDE diffusion is isotropic, or its matrix decomposition is known.

Acknowledgements

This work is supported by the Air Force Office of Scientific Research, Grant FA 9550-09-1-0047.

References

[1] S. Asmussen, and P. W. Glynn *Stochastic Simulation: Algorithms and Analysis*, Springer, 2007

[2] N. Bou-Rabee, and E. Vanden-Eijnden, *Pathwise Accuracy and Ergodicity of Metropolized Integrators for SDEs*, Comm. Pure Appl. Math. Vol. 63, No. 5, (2010), pp. 655-696

[3] R. Cao, and S. B. Pope, *Numerical Integration of Stochastic Differential Equations: Weak Second-Order Mid-Point Scheme for Application in the Composition PDF Method*, J. Comput. Phys. **185**, (2003), pp. 194-212

[4] P.E. Kloeden, and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Springer, Berlin, 1992

[5] M.B. Giles, *Multilevel Monte Carlo path simulation*, Oper. Res. Vol. 56, No. 3 (2008), pp. 607-617

[6] M.B. Giles, D.J. Higham, and X. Mao *Analysing Multi-Level Monte Carlo for Options with Non-globally Lipschitz Payoff*, Finance Stoch. Vol. 13, No. 3 (2009), pp. 403-413

[7] G.N. Milstein, and M.V. Tretyakov, *Stochastic Numerics for Mathematical Physics*, Springer, 2004

[8] P. P. Popov, and S. B. Pope *Stochastic Particle Advection in Hybrid Large Eddy Simulation/Filtered Density Function Methods*, B. Am. Phys. Soc., 61st Annual Meeting of the APS Division of Fluid Dynamics, Vol. 53, No. 15, San Antonio, Texas, Nov 23-25, 2008

[9] P. Protter, and D. Talay, *The Euler Scheme for Levy Driven Stochastic Differential Equations*, Ann. Probab. Vol. 25, No. 1, (1997), pp. 393–423

[10] L. F. Richardson, and J. A. Gaunt, *The Deferred Approach to the Limit. Part I. Single Lattice. Part II. Interpenetrating Lattices*, Philos. T. Roy. Soc. A, **226**, (1927), pp. 299-361

[11] D. Talay, and L. Tubaro *Expansion of the Global Error for Numerical Schemes Solving Stochastic Differential Equations*, Stoch. Anal. Appl., **8** 94-120 (1990)

[12] A. Tocino, and J. Vigo-Aguiar, *Weak Second Order Conditions for Stochastic Runge-Kutta Methods*, SIAM J. Sci. Comput., Vol. 24, No. 2, (2002), pp. 507–523

[13] H. Wang, P. P. Popov, and S. B. Pope *Weak Second Order Splitting Schemes for LAgrangian Monte Carlo Particle Methods for the Composition PDF/FDF Transport Equations*, J. Comput. Phys., **229**, 2010, pp. 1852-1878,

Appendix 1: Proof of Theorem 1

Here, we prove Theorem 1.

Theorem 1: For any integer $m, p \geq 1$, and any real number $d > 0$, if the fields $\mathbf{D}(\mathbf{x}, t)$ and $\sigma(\mathbf{x}, t)$ satisfy the regularity conditions stated in eqs.(25 - 32), then there exists a constant $\bar{C} \in (0, \infty)$, dependent only on $p, m, d, \{C\}$, such that the set $E \subset R^{n2^{p-1}}$, as defined by

$$E = \left\{ (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}}) \in R^{n2^{p-1}} \left| \left| \frac{f_{\Delta t}^{U,k}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})}{f_{\Delta t}^{U,p-1}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})} - 1 \right| > d \right. \right\}, \tag{60}$$

satisfies

$$\int_E f_{\Delta t}^{U,k}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)\left|\mathrm{d}\mathbf{x}_1\right|\left|\mathrm{d}\mathbf{x}_2\right|...\left|\mathrm{d}\mathbf{x}_{2^{p-1}}\right| \leq \bar{C}\Delta t^m \tag{61}$$

and

$$\int_E f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)\left|\mathrm{d}\mathbf{x}_1\right|\left|\mathrm{d}\mathbf{x}_2\right|...\left|\mathrm{d}\mathbf{x}_{2^{p-1}}\right| \leq \bar{C}\Delta t^m \tag{62}$$

Proof of Theorem 1: We use the convention $\mathbf{x}_0 = \mathbf{0}$. For a given value of $\epsilon \in (0,\infty)$, consider the set $G_{\Delta t}^{\epsilon} \in R^{n2^{p-1}}$ defined by:

$$G_{\Delta t}^{\epsilon} = \left\{\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right) \in R^{n2^{p-1}} \left| \left\|\mathbf{x}_j - \mathbf{x}_{j-1}\right\| \leq \Delta t^{1/2-\epsilon}\right.\right\} \tag{63}$$

Our proof consists of two parts. In the first, we demonstrate that for a suitable ϵ and a small enough Δt, a point $(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})$ is in E only if it is also in the complement of $G_{\Delta t}^{\epsilon}$ (in other words $E \subseteq \left(G_{\Delta t}^{\epsilon}\right)^c$). In the second part, we prove that the integral of either $f_{\Delta t}^{U,k}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ or $f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ over the set $\left(G_{\Delta t}^{\epsilon}\right)^c$ is small - the reader can get some intuition as to why this is so by noting that, by their definition, the functions $f_{\Delta t}^{U,k}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ and $f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ consist of products of Gaussian PDFs with charateristic width $\Delta t^{1/2}$, whereas $G_{\Delta t}^{\epsilon}$ is a region of characteristic width $\Delta t^{1/2-\epsilon}$, and for small Δt we have that $\Delta t^{1/2-\epsilon} > \Delta t^{1/2}$, so that $G_{\Delta t}^{\epsilon}$ contains most of the mass of $f_{\Delta t}^{U,k}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$ and $f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right)$.

Consider a given point $\left(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}}\right) \in G_{\Delta t}^{\epsilon}$. We have then, from eqs.(35 -39), that

$$\begin{aligned}\frac{f_{\Delta t,U_j}^{U,k}\left(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1}\right)}{f_{\Delta t,U_j}^{U,p-1}\left(\mathbf{x}_j;\mathbf{x}_2,...,\mathbf{x}_{j-1}\right)} \quad &= \\ &\frac{\left|B\left(\mathbf{x}_{j-1},t_{j-1}\right)\frac{\Delta t}{2^{p-1}}\right|^{1/2}}{\left|B\left(\mathbf{x}^*,t^*\right)\frac{\Delta t}{2^{p-1}}\right|^{1/2}} \\ &\times\frac{\exp\left(-\mathbf{v}^T\frac{1}{2}\left(B\left(\mathbf{x}^*,t^*\right)\frac{\Delta t}{2^{p-1}}\right)^{-1}\mathbf{v}\right)}{\exp\left(-\tilde{\mathbf{v}}^T\frac{1}{2}\left(B\left(\mathbf{x}_{j-1},t_{j-1}\right)\frac{\Delta t}{2^{p-1}}\right)^{-1}\tilde{\mathbf{v}}\right)},\end{aligned} \tag{64}$$

where $\mathbf{x}^*,\mathbf{v},t^*$ are as defined in eqs.(38, 39), and we introduce, for the sake of compactness, the notation $t_{j-1} = (j-1)\frac{\Delta t}{2^{p-1}}$, and we shall also denote $\tilde{\mathbf{v}} = \left[\mathbf{x}_j - \mathbf{x}_{j-1} - \mathbf{D}\left(\mathbf{x}_{j-1},t_{j-1}\right)\frac{\Delta t}{2^{p-1}}\right]$.

Considering the term $\frac{\left|B(\mathbf{x}_{j-1},t_{j-1})\frac{\Delta t}{2^{p-1}}\right|^{1/2}}{\left|B(\mathbf{x}^*,t^*)\frac{\Delta t}{2^{p-1}}\right|^{1/2}}$ first, we have that since $\left\|\mathbf{x}_{j-1}-\mathbf{x}^*\right\| \leq \Delta t^{1/2-\epsilon}$ and $\left|t_{j-1}-t^*\right| \leq \Delta t$, then from the regularity equations, eqs.(25 - 32),

$$\left|\frac{\left|B\left(\mathbf{x}_{j-1},t_{j-1}\right)\frac{\Delta t}{2^{p-1}}\right|^{1/2}}{\left|B\left(\mathbf{x}^{*},t^{*}\right)\frac{\Delta t}{2^{p-1}}\right|^{1/2}}-1\right|\leq\left(\frac{C_{3}^{\sigma,\mathbf{x}}\Delta t^{1/2-\epsilon}+C_{3}^{\sigma,t}\Delta t}{C_{2}^{\sigma}}\right)^{n} \tag{65}$$

Next, considering the term $\frac{\exp\left(-\mathbf{v}^{T}\frac{1}{2}\left(B(\mathbf{x}^{*},t^{*})\frac{\Delta t}{2^{p-1}}\right)^{-1}\mathbf{v}\right)}{\exp\left(-\tilde{\mathbf{v}}^{T}\frac{1}{2}\left(B(\mathbf{x}_{j-1},t_{j-1})\frac{\Delta t}{2^{p-1}}\right)^{-1}\tilde{\mathbf{v}}\right)}$, we have that

$$\frac{\exp\left(-\mathbf{v}^{T}\frac{1}{2}\left(B\left(\mathbf{x}^{*},t^{*}\right)\frac{\Delta t}{2^{p-1}}\right)^{-1}\mathbf{v}\right)}{\exp\left(-\tilde{\mathbf{v}}^{T}\frac{1}{2}\left(B\left(\mathbf{x}_{j-1},t_{j-1}\right)\frac{\Delta t}{2^{p-1}}\right)^{-1}\tilde{\mathbf{v}}\right)}=\exp\left(-\frac{2^{p-2}}{\Delta t}\left(\mathbf{w}^{T}\mathbf{w}-\tilde{\mathbf{w}}^{T}\tilde{\mathbf{w}}\right)\right), \tag{66}$$

where $\mathbf{w}=\sigma\left(\mathbf{x}^{*},t^{*}\right)^{-1}\mathbf{v}$ and $\tilde{\mathbf{w}}=\sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\tilde{\mathbf{v}}$. Now, we have that $\mathbf{w}^{T}\mathbf{w}-\tilde{\mathbf{w}}^{T}\tilde{\mathbf{w}}=\left(\mathbf{w}-\tilde{\mathbf{w}}\right)^{T}\left(\mathbf{w}+\tilde{\mathbf{w}}\right)$, and using again eqs.(25 - 32), we have that

$$\left\|\mathbf{w}+\tilde{\mathbf{w}}\right\|\leq2C_{1}^{\sigma^{-1}}\left(\Delta t^{1/2-\epsilon}+C_{1}^{D}\frac{\Delta t}{2^{p-1}}\right), \tag{67}$$

and

$$\begin{aligned}\mathbf{w}-\tilde{\mathbf{w}} &= \sigma\left(\mathbf{x}^{*},t^{*}\right)^{-1}\mathbf{v}-\sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\tilde{\mathbf{v}}\\ &= \left(\sigma\left(\mathbf{x}^{*},t^{*}\right)^{-1}-\sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\right)\mathbf{v}+\sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\left(\mathbf{v}-\tilde{\mathbf{v}}\right)\\ &= \left(\sigma\left(\mathbf{x}^{*},t^{*}\right)^{-1}-\sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\right)\mathbf{v}\\ &+ \sigma\left(\mathbf{x}_{j-1},t_{j-1}\right)^{-1}\left(\mathbf{D}\left(\mathbf{x}_{j-1},t_{j-1}\right)-\mathbf{D}\left(\mathbf{x}^{*},t^{*}\right)\right)\frac{\Delta t}{2^{p-1}}.\end{aligned} \tag{68}$$

Again invoking eqs.(25 - 32) to bound the magnitude of the two terms on the rightmost side of eq.(68), we have that

$$\begin{aligned}\left\|\mathbf{w}-\tilde{\mathbf{w}}\right\| &\leq \left(\Delta t^{1/2-\epsilon}+C_{1}^{D}\frac{\Delta t}{2^{p-1}}\right)\left(C_{3}^{\sigma^{-1},\mathbf{x}}2^{p-1}\Delta t^{1/2-\epsilon}+C_{3}^{\sigma^{-1},t}\Delta t\right)\\ &+ \frac{\Delta t}{2^{p-1}}C_{1}^{\sigma^{-1}}\left(C_{2}^{D,\mathbf{x}}2^{p-1}\Delta t^{1/2-\epsilon}+C_{2}^{D,t}\Delta t\right).\end{aligned} \tag{69}$$

Combining the results of eqs.(67, 69), we get that for $\Delta t<1,\epsilon\in(0,0.5)$ there exists a constant $A_{1}\in(0,\infty)$ such that

$$\left\|\mathbf{w}^{T}\mathbf{w}-\tilde{\mathbf{w}}^{T}\tilde{\mathbf{w}}\right\|\leq A_{1}\Delta t^{3/2-3\epsilon}. \tag{70}$$

Choosing $\epsilon=1/12$, substituting the result of eq.(70) into eq.(66), and combining with the result of eq.(65) we get that for any $h\in(0,\infty)$, we can choose Δt small enough such that if $(\mathbf{x}_{1},\mathbf{x}_{2},...,\mathbf{x}_{2^{p-1}})\in G_{\Delta t}^{1/12}$, then we have that $\left|\frac{f_{\Delta t,U_{j}}^{U,k}(\mathbf{x}_{j};\mathbf{x}_{1},...,\mathbf{x}_{j-1})}{f_{\Delta t,U_{j}}^{U,p-1}(\mathbf{x}_{j};\mathbf{x}_{2},...,\mathbf{x}_{j-1})}-1\right|\leq h$. Since this result holds for any $j\in\left\{1,2,3,...,2^{p-1}\right\}$,

by eq.(36) we can choose h small enough that $\left|\frac{f_{\Delta t}^{U,k}\left(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_{2^{p-1}}\right)}{f_{\Delta t}^{U,p-1}\left(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_{2^{p-1}}\right)}-1\right|\leq d$ for any $(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\in G_{\Delta t}^{1/12}$.

Therefore, for Δt small enough, we have that $E\subseteq\left(G_{\Delta t}^{1/12}\right)^c$ which implies that

$$\begin{aligned}&\int_E f_{\Delta t}^{U,k}(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|\\&\leq\int_{\left(G_{\Delta t}^{1/12}\right)^c} f_{\Delta t}^{U,k}(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|\end{aligned}\tag{71}$$

Note that if $(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\in\left(G_{\Delta t}^{1/12}\right)^c$, then for some i between 1 and 2^{p-1} we have that $\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}$, and so we have that

$$\begin{aligned}&\int_{\left(G_{\Delta t}^{1/12}\right)^c} f_{\Delta t}^{U,k}(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|\\&\leq\sum_{i=1}^{2^{p-1}}\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}} f_{\Delta t}^{U,k}(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|,\end{aligned}\tag{72}$$

where $\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}$ denotes the set of all points $(\mathbf{x}_1,\mathbf{x}_2,...\mathbf{x}_{2^{p-1}})\in R^{n2^{p-1}}$ such that $\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}$. Let's consider a single term in the sum in eq.(72). We have, by eq.(35), that

$$\begin{aligned}&\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}} f_{\Delta t}^{U,k}(\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|\\&=\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}}\prod_{j=1}^{2^{p-1}} f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_{2^{p-1}}|\\&=\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}}\prod_{j=1}^{i} f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_i|,\end{aligned}\tag{73}$$

where the second inequality follows from Fubini's theorem and the fact that, since $f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})$ are conditional PDFs, as defined in eq.(37) then for any $\mathbf{x}_1,...,\mathbf{x}_{j-1}$ we have that

$$\int_{R^n} f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})\,|\mathrm{d}\mathbf{x}_j|=1.\tag{74}$$

By inductively applying eq.(74), we also get that

$$\int_{R^{n(i-1)}}\prod_{j=1}^{i-1} f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})\,|\mathrm{d}\mathbf{x}_1|\,|\mathrm{d}\mathbf{x}_2|\ldots|\mathrm{d}\mathbf{x}_{i-1}|=1.\tag{75}$$

Applying eq.(75) and Fubini's theorem to eq.(73), we get that

$$\begin{aligned}&\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}}\prod_{j=1}^{i} f_{\Delta t,U_j}^{U,k}(\mathbf{x}_j;\mathbf{x}_1,...,\mathbf{x}_{j-1})\,|\mathrm{d}\mathbf{x}_1|\ldots|\mathrm{d}\mathbf{x}_i|\\&\leq\sup_{(\mathbf{x}_1,\ldots,\mathbf{x}_{i-1})}\left(\int_{\left\{\|\mathbf{x}_i-\mathbf{x}_{i-1}\|\geq\Delta t^{5/12}\right\}} f_{\Delta t,U_i}^{U,k}(\mathbf{x}_i;\mathbf{x}_1,...,\mathbf{x}_{i-1})\,|\mathrm{d}\mathbf{x}_i|\right),\end{aligned}\tag{76}$$

where the supremum in the above inequality is taken over all $(\mathbf{x}_1, ..., \mathbf{x}_{i-1}) \in R^{n(i-1)}$. Note, however, that by its definition (eq.(37)), we have that the function $f^{U,k}_{\Delta t,U_i}(\mathbf{x}_i; \mathbf{x}_1, ..., \mathbf{x}_{i-1})$ is a multivariate Gaussian distribution whose covariance matrix roughly scales as $\frac{\Delta t}{2^{p-1}}$, and so it is intuitively easy to see that integrating that distribution over the region $\left\{\|\mathbf{x}_i - \mathbf{x}_{i-1}\| \geq \Delta t^{5/12}\right\}$ gives a quantity which decreases faster than any power of Δt, as $\Delta t \downarrow 0$. More concretely, we have that, for fixed $\mathbf{x}_1, ..., \mathbf{x}_{i-1}$, using the definition of $f^{U,k}_{\Delta t,U_i}$ and the bounds given by eqs.(25 - 32), the following inequality holds

$$f^{U,k}_{\Delta t,U_i}(\mathbf{x}_i; \mathbf{x}_1, ..., \mathbf{x}_{i-1}) \leq \tag{77}$$
$$\frac{1}{\left(2\pi\frac{\Delta t}{2^{p-1}}\right)^{n/2}\left(C_2^\sigma\right)^n} \exp\left(-\frac{\left(C_2^{\sigma^{-1}}\right)^2 2^{p-2}}{\Delta t}\left\|\mathbf{x}_i - \mathbf{x}_{i-1} - \mathbf{D}(\mathbf{x}^*, t^*)\frac{\Delta t}{2^{p-1}}\right\|^2\right),$$

and for $\Delta t \leq \left(\frac{2^{p-2}}{C_1^D}\right)^{12/7}$, we have that $\|\mathbf{x}_i - \mathbf{x}_{i-1}\| \geq \Delta t^{5/12}$ implies that $\left\|\mathbf{x}_i - \mathbf{x}_{i-1} - \mathbf{D}(\mathbf{x}^*, t^*)\frac{\Delta t}{2^{p-1}}\right\| \geq \frac{1}{2}\|\mathbf{x}_i - \mathbf{x}_{i-1}\|$, hence for $\Delta t \leq \left(\frac{2^{p-2}}{C_1^D}\right)^{12/7}$ and $\|\mathbf{x}_i - \mathbf{x}_{i-1}\| \geq \Delta t^{5/12}$, we have that

$$f^{U,k}_{\Delta t,U_i}(\mathbf{x}_i; \mathbf{x}_1, ..., \mathbf{x}_{i-1}) \leq \tag{78}$$
$$\frac{1}{\left(2\pi\frac{\Delta t}{2^{p-1}}\right)^{n/2}\left(C_2^\sigma\right)^n} \exp\left(-\frac{\left(C_2^{\sigma^{-1}}\right)^2 2^{p-4}}{\Delta t}\|\mathbf{x}_i - \mathbf{x}_{i-1}\|^2\right),$$

which implies that

$$\sup_{(\mathbf{x}_1,...,\mathbf{x}_{i-1})}\left(\int_{\left\{\|\mathbf{x}_i - \mathbf{x}_{i-1}\| \geq \Delta t^{5/12}\right\}} f^{U,k}_{\Delta t,U_i}(\mathbf{x}_i; \mathbf{x}_1, ..., \mathbf{x}_{i-1})\,|\mathrm{d}\mathbf{x}_i|\right)$$
$$\leq \int_{\left\{\|\mathbf{y}\| \geq \Delta t^{5/12}\right\}} \frac{1}{\left(2\pi\frac{\Delta t}{2^{p-1}}\right)^{n/2}\left(C_2^\sigma\right)^n} \exp\left(-\frac{\left(C_2^{\sigma^{-1}}\right)^2 2^{p-4}}{\Delta t}\|\mathbf{y}\|^2\right)|\mathrm{d}\mathbf{y}|$$
$$= \int_{\left\{\|\mathbf{z}\| \geq \Delta t^{-1/12}\right\}} \frac{1}{\left(\frac{2\pi}{2^{p-1}}\right)^{n/2}\left(C_2^\sigma\right)^n} \exp\left(-\left(C_2^{\sigma^{-1}}\right)^2 2^{p-4}\|\mathbf{z}\|^2\right)|\mathrm{d}\mathbf{z}|, \tag{79}$$

where the third line follows from the second by the simple change of variables $\mathbf{z} = \mathbf{y}\Delta t^{-1/2}$. Using this result in conjunction with eqs.(76, 73, 72), we get that for $\Delta t \leq \left(\frac{2^{p-2}}{C_1^D}\right)^{12/7}$

$$\int_{\left(G^{1/12}_{\Delta t}\right)^c} f^{U,k}_{\Delta t}(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{2^{p-1}})\,|\mathrm{d}\mathbf{x}_1|...|\mathrm{d}\mathbf{x}_{2^{p-1}}|$$
$$\leq 2^{p-1}\int_{\left\{\|\mathbf{z}\| \geq \Delta t^{-1/12}\right\}} \frac{1}{\left(\frac{2\pi}{2^{p-1}}\right)^{n/2}\left(C_2^\sigma\right)^n} \exp\left(-\left(C_2^{\sigma^{-1}}\right)^2 2^{p-4}\|\mathbf{z}\|^2\right)|\mathrm{d}\mathbf{z}|, \tag{80}$$

and it is a simple, albeit lengthy, Calculus exercise to demonstrate that for any integer m there exists a constant $\bar{C}$ such that

$$2^{p-1}\int_{\left\{\|\mathbf{z}\|\geq\Delta t^{-1/12}\right\}}\frac{1}{\left(\frac{2\pi}{2^{p-1}}\right)^{n/2}\left(C_2^{\sigma}\right)^n}\exp\left(-\left(C_2^{\sigma^{-1}}\right)^2 2^{p-4}\left\|\mathbf{z}\right\|^2\right)|\mathrm{d}\mathbf{z}|\leq\bar{C}\Delta t^m, \tag{81}$$

which proves the first part of Theorem 1, eq.(61). The proof of the second part is completely identical, with the only difference being that we have $(\mathbf{x}_{i-1},t_{i-1})$ in place of $(\mathbf{x}^*,t^*)$ in eq.(77), which has no impact on the argument of eqs.(77 - 81). And so, we have proven Theorem 1.

Appendix 2: Proof of Theorem 2

Here, we prove Theorem 2, which was used to demonstrate that the DRp schemes satisfy the criterion for weak p-th order accuracy.

Theorem 2: For the random variable $\mathbf{Z}_{\Delta t}^{DRp}$ defined in section 7, if eq.(23) holds, then so also does eq.(24).

Proof of Theorem 2: We shall divide the domain, $\mathbb{R}^n$, into two parts: a ball of radius 1 centered on the origin, $\mathcal{B}(\mathbf{0},1)$, and its complement, $\mathcal{B}(\mathbf{0},1)^c$. We have that

$$\begin{aligned}
\left|E\left(\prod_{m=1}^{n}\left(Z_{\Delta t,m}^{DRp}\right)^{i_m}-\sum_{k=1}^{p}l_k^p\prod_{m=1}^{n}\left(Z_{\frac{\Delta t}{2^{k-1}},m}^{E}\right)^{i_m}\right)\right| &= \\
\left|\int\prod_{m=1}^{n}x_m^{i_m}\left(f_{\Delta t}^{DRp}(\mathbf{x})-\sum_{k=0}^{p-1}l_{k+1}^p f_{\Delta t}^k(\mathbf{x})\right)|\mathrm{d}\mathbf{x}|\right| &\leq \\
\int_{\mathcal{B}(\mathbf{0},1)}\|\mathbf{x}\|^{\sum_{m=1}^n i_m}\left|f_{\Delta t}^{DRp}(\mathbf{x})-\sum_{k=0}^{p-1}l_{k+1}^p f_{\Delta t}^k(\mathbf{x})\right||\mathrm{d}\mathbf{x}| &+ \\
\int_{\mathcal{B}(\mathbf{0},1)^c}\|\mathbf{x}\|^{\sum_{m=1}^n i_m}\left|f_{\Delta t}^{DRp}(\mathbf{x})-\sum_{k=0}^{p-1}l_{k+1}^p f_{\Delta t}^k(\mathbf{x})\right||\mathrm{d}\mathbf{x}|\,, &
\end{aligned} \tag{82}$$

and since $\|\mathbf{x}\|^{\sum_{m=1}^n i_m}$ is bounded on $\mathcal{B}(\mathbf{0},1)$, we have that eq.(23) implies that there exists a constant, C'', such that

$$\int_{\mathcal{B}(\mathbf{0},1)}\|\mathbf{x}\|^{\sum_{m=1}^n i_m}\left|f_{\Delta t}^{DRp}(\mathbf{x})-\sum_{k=0}^{p-1}l_{k+1}^p f_{\Delta t}^k(\mathbf{x})\right||\mathrm{d}\mathbf{x}|\leq C''\Delta t^{p+1} \tag{83}$$

Now, let us consider the second term in eq.(82), and obtain a result analogous to eq.(83) for it. From the boundedness of σ and $\mathbf{D}$, there exists a parameter

Δt_0, dependent on $\{C\}$ and n only, such that for all $\Delta t \leq \Delta t_0$, and any k between 0 and $p-1$, we have that $\|\mathbf{x}\| \geq 1$ implies that

$$f_{\Delta t}^{k}(\mathbf{x}) \leq \frac{1}{(2\pi\Delta t)^{n/2}\left(2C_1^{\sigma}\right)^{n}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2\Delta t\left(2C_1^{\sigma}\right)^2}\right). \tag{84}$$

Also, since $c > 0$, as defined in Section 7, gives a lower bound on the acceptance probability, we have that

$$\frac{f_{\Delta t}^{W,p-1}}{f_{\Delta t}^{U,p-1}} \leq \frac{1}{c}, \tag{85}$$

which implies that

$$\frac{f_{\Delta t}^{DRp}}{f_{\Delta t}^{p-1}} \leq \frac{1}{c}, \tag{86}$$

and so, eqs.(84,86) jointly imply that for $\Delta t \leq \Delta t_0$ and $\mathbf{x} \in \mathcal{B}(\mathbf{0},1)^c$, we have that

$$\begin{aligned} \left| f_{\Delta t}^{DRp}(\mathbf{x}) - \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{k}(\mathbf{x}) \right| \quad \leq \\ \left(\frac{1}{c} + \sum_{k=0}^{p-1} \left|l_{k+1}^{p}\right|\right) \frac{1}{(2\pi\Delta t)^{n/2}\left(2C_1^{\sigma}\right)^{n}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2\Delta t\left(2C_1^{\sigma}\right)^2}\right), \end{aligned} \tag{87}$$

which implies that for $\Delta t \leq \Delta t_0$ we have

$$\begin{aligned} \int_{\mathcal{B}(\mathbf{0},1)^c} \|\mathbf{x}\|^{\sum_{m=1}^{n} i_m} \left| f_{\Delta t}^{DRp}(\mathbf{x}) - \sum_{k=0}^{p-1} l_{k+1}^{p} f_{\Delta t}^{k}(\mathbf{x}) \right| |\mathrm{d}\mathbf{x}| \quad \leq \\ \int_{\mathcal{B}(\mathbf{0},1)^c} \|\mathbf{x}\|^{\sum_{m=1}^{n} i_m} \frac{\frac{1}{c} + \sum_{k=0}^{p-1} \left|l_{k+1}^{p}\right|}{(2\pi\Delta t)^{n/2}\left(2C_1^{\sigma}\right)^{n}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2\Delta t\left(2C_1^{\sigma}\right)^2}\right) |\mathrm{d}\mathbf{x}| \end{aligned} \tag{88}$$

As the Gaussian distribution on the second line of eq.(88) has characteristic width $2C_1^{\sigma}\sqrt{\Delta t}$, whereas the domain of integration is over all $\mathbf{x}$ with $\|\mathbf{x}\| \geq 1$, it is a simple, albeit lengthy, Calculus exercise to demonstrate that there exists a constant C'' such that

$$\int_{\mathcal{B}(\mathbf{0},1)^c} \|\mathbf{x}\|^{\sum_{m=1}^{n} i_m} \frac{\frac{1}{c} + \sum_{k=0}^{p-1} \left|l_{k+1}^{p}\right|}{(2\pi\Delta t)^{n/2}\left(2C_1^{\sigma}\right)^{n}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2\Delta t\left(2C_1^{\sigma}\right)^2}\right) |\mathrm{d}\mathbf{x}| \leq C''\Delta t^{p+1}. \tag{89}$$

Combining the results of eqs.(83,89) into eq.(82), eq.(24) follows immediately, which concludes the proof of Theorem 2.

Specific Volume Coupling and Convergence Properties in Hybrid Particle/Finite Volume Algorithms for Turbulent Reactive Flows

Pavel P. Popov Haifeng Wang
Stephen B. Pope
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

March 29, 2012

Abstract

We investigate the coupling between the two components of a Large Eddy Simulation / Probability Density Function (LES/PDF) algorithm for the simulation of turbulent reacting flows. In such an algorithm, the Large Eddy Simulation (LES) component provides a solution to the hydrodynamic equations, whereas the Lagrangian Monte Carlo Probability Density Function (PDF) component solves for the PDF of chemical compositions. Special attention is paid to the transfer of specific volume information from the PDF to the LES code: the specific volume field contains probabilistic noise due to the nature of the Monte Carlo PDF solution, and thus the use of the specific volume field in the LES pressure solver needs careful treatment. Using a test flow based on the Sandia/Sydney Bluff Body Flame, we determine the optimal strategy for specific volume feedback. Then, the overall second-order convergence of the entire LES/PDF procedure is verified using a simple vortex ring test case, with special attention being given to bias errors due to the number of particles per LES finite volume (FV) cell.

1 Introduction

The PDF method for the treatment of turbulent reactive flows [1,2] has been shown to be highly effective, due to the fact that the chemical source term, which is highly non-linear in a combustion application, appears in closed form and thus requires no modeling [3]. Initially, the PDF methodology was coupled to Reynolds-averaged Navier-Stokes (RANS) hydrodynamic solvers, thus

giving rise to the RANS/PDF class of algorithms for simulation of turbulent reactive flows, which are to this date effectively used in simulating combustion processes (see [3] for a review). We note that there are three main classes of computationally tractable algorithms for approximating the composition PDF [3]. Here we focus on the Lagrangian Particle Monte Carlo class, in which the PDF is approximated from an ensemble of points (from here on referred to as "particles") that are advected and diffused in physical space according to the LES resolved velocity and turbulent diffusivity [1,9] (in this paper, we use the terms "turbulent viscosity" and "turbulent diffusivity" to denote the viscosity and diffusivity used to model the turbulent motions unresolved by the LES grid). In the related Eulerian Particle Monte Carlo method [22], the locations of particles are fixed to the grid nodes. Yet another class of PDF algorithms, the Eulerian Field Monte Carlo class [21,20,8], instead employs an ensemble of composition fields defined over the entire domain, which evolve by partial differential equations with a stochastic forcing component. All of these approaches have their strengths and differences [3]: for example, the Lagrangian Particle Monte Carlo approach allows for more accurate treatment of sub-filter mixing and diffusion [6]. With increasing computational resources, the Large Eddy Simulation (LES) approach for turbulence modeling has been supplanting RANS as the hydrodynamic solver used for turbulent combustion simulations [17]. First introduced by Pope [16], hybrid Large Eddy Simulation / Probability Density Function (LES/PDF) methods have the added benefit that the LES approach eliminates the need for modeling of the large scale turbulent motions, which are highly geometry-dependent and fall outside of the scope of the Kolmogorov hypotheses [2]. Hybrid LES/PDF methods have been demonstrated to be highly effective in simulating laboratory-scale flames [9, 10, 11, 18].

In a typical LES/PDF simulation, the LES code provides fields of velocity and turbulent diffusivity to the PDF code. The PDF code then transports the particles in physical space, performs molecular diffusion and mixing, and chemical reaction steps, and then passes back to the LES code the fields of specific volume, molecular viscosity and diffusivity. The very different nature of the two components of an LES/PDF algorithm poses serious challenges for the implementation of information transfer between the stochastic PDF code, whose fields contain considerable random errors, and the LES code, which is a straightforward finite volume (FV) algorithm, employing discretization schemes which assume a certain level of regularity from the fields on which they operate. Here, we examine in detail this interplay between the LES and PDF components. Particular emphasis is given to the feedback of specific volume from PDF to LES. In advancing the hydrodynamic variables, the LES code solves a Poisson equation for pressure, whose source term contains the rate of change of specific volume. Since specific volume information is obtained from the stochastic PDF code, special effort must be made to ensure that the specific volume fields which are input into the LES pressure solver contain as little statistical error as possible. Furthermore, it is desired that the overall LES/PDF time stepping procedure be second-order accurate in space and time. In the context of the LES/PDF code developed by the Turbulence and Combustion Group at Cornell University [18],

we address these two issues, examining different strategies in performing PDF to LES feedback, and testing the overall order of convergence of the LES/PDF code.

We note that specific volume feedback from PDF to LES (or from PDF to RANS, which is a closely related problem) has previously been addressed as a component of an overall algorithm for the simulation of turbulent reactive flows. Previous researchers have proposed specific volume coupling schemes based on direct evaluation from the particle ensemble [10] (analogous to the scheme PSV described in Section 3), and schemes based on extracting specific volume information from an auxiliary transported scalar (analogous to the scheme TSV described in Section 3), where the transported scalar is either enthalpy [13, 12, 4, 11] or temperature [10]. While using a similar formulation at the PDE level for specific volume feedback, the present work extends the above studies by examining the LES to PDF coupling error in detail, isolating it from other sources of numerical errors in an LES/PDF algorithm, and determining the optimal coupling strategy. A further original contribution of the present work is in the development of an LES/PDF coupling algorithm which allows for second-order accuracy of the overall code (with respect to both the grid size and time step); this second-order convergence is then verified numerically.

The rest of this paper is organized as follows: in Section 2, we present the equations solved by an LES/PDF algorithm for turbulent reactive flows. Section 3 describes in detail the issues that arise in the coupling of the LES and PDF algorithms; two alternative approaches for specific volume coupling are outlined. The numerical algorithms for coupling between the LES and PDF codes are described in Section 4. Numerical tests of these couplings are performed in Sections 5 and 6, in which the fully-coupled LES/PDF implementations are tested against a standalone-LES (S-LES) code with no coupling errors. Section 5 describes results from a long time interval turbulent jet flow, representative of a typical LES/PDF simulation. In Section 6, we show convergence results from a simple vortex ring numerical test case, which demonstrates that the overall LES/PDF algorithm is second-order accurate in space and time.

2 Equations Solved by the LES/PDF Algorithm

In this section, we present the equations which are solved by an LES/PDF algorithm. Throughout the present paper we use a tilde to denote mass weighted resolved quantities (also referred to here as Favre-averaged quantities) solved for by the LES/PDF solver, e.g. $\widetilde{\mathbf{u}}$ for the mass-weighted resolved velocity vector, and $\widetilde{\boldsymbol{\phi}}$ for the mass-weighted resolved composition vector. Additionally, the composition PDF solver computes the evolution of individual particle properties, which we denote by a star superscript, e.g. ϕ^*_α for the $\alpha-$component of an individual particle's composition vector.

In the present study, we use a simple flamelet chemistry model [19], and the composition vector ϕ_α consists of a single component, the mixture fraction. This still allows for considerable specific volume variation throughout the flow,

which is necessary for testing the specific volume coupling. Furthermore, using a mixture-fraction-only composition vector enables us to perform simulations of the same flow via both LES/PDF and standalone-LES methods, with both solutions evolving by the same governing equations. This allows us to approximate the LES/PDF coupling error as the difference between the standalone-LES and the LES/PDF solutions.

First, we describe the governing equations for a standalone-LES simulation with flamelet chemistry modeling. The LES governing equations are the same as those used by Pierce and Moin [7] (with the exception that in [7] the authors also use a progress variable in the flamelet model), by Wang and Pope [18], and by Kemenov et al. [19].

2.1 Governing equations for a standalone-LES simulation

We denote by $\widetilde{u}_j, \widetilde{\xi}, \widetilde{\xi^2}$, respectively, the Favre-averaged LES velocity, mixture fraction and square of mixture fraction. The Favre-averaged molecular viscosity and diffusivity are denoted as $\widetilde{\nu}$ and $\widetilde{D}$, and their turbulent counterparts are $\widetilde{\nu}_T, \widetilde{D}_T$ respectively. The turbulent viscosity, $\widetilde{\nu}_T$, is evaluated by the dynamic Smagorinsky procedure, and the turbulent diffusivity, $\widetilde{D}_T$, is determined by using a specified value for the turbulent Schmidt number:

$$\frac{\widetilde{\nu}_T}{\widetilde{D}_T} = \sigma_T, \ \sigma_T = 0.4. \tag{1}$$

Finally, we denote with $\overline{p}, \overline{\rho}$, respectively, the LES resolved pressure and density. With these definitions, the variable-density incompressible standalone-LES code solves the following set of equations

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u}_j}{\partial x_j} = 0, \tag{2}$$

$$\frac{\partial \left(\overline{\rho}\widetilde{u}_j\right)}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u}_i\widetilde{u}_j)}{\partial x_i} = -\frac{\partial \overline{p}}{\partial x_j} + 2\frac{\partial}{\partial x_i}\left(\overline{\rho}\left(\widetilde{\nu}+\widetilde{\nu}_T\right)\left(\widetilde{S}_{ij} - \frac{1}{3}\widetilde{S}_{kk}\delta_{ij}\right)\right), \tag{3}$$

$$\frac{\partial \left(\overline{\rho}\widetilde{\xi}\right)}{\partial t} + \frac{\partial \left(\overline{\rho}\widetilde{u}_j\widetilde{\xi}\right)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\overline{\rho}\left(\widetilde{D}+\widetilde{D}_T\right)\frac{\partial \widetilde{\xi}}{\partial x_j}\right), \tag{4}$$

$$\frac{\partial \left(\overline{\rho}\widetilde{\xi^2}\right)}{\partial t} + \frac{\partial \left(\overline{\rho}\widetilde{u}_j\widetilde{\xi^2}\right)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\overline{\rho}\left(\widetilde{D}+\widetilde{D}_T\right)\frac{\partial \widetilde{\xi^2}}{\partial x_j}\right) + S_{\xi^2}. \tag{5}$$

Equations (2-5) are respectively the Favre-averaged forms of the continuity, momentum, and scalar evolution equations for the mixture fraction, ξ, and its square ξ^2. The tensor $\widetilde{S}_{ij}$ is the resolved strain rate, whereas the scalar S_{ξ^2} is a source term in the scalar evolution equation for $\widetilde{\xi^2}$, determined by the scalar mixing model. For the present study, which uses the dynamic Smagorinsky procedure, we denote by Δ the Smagorinsky filter size, and the exact form of S_{ξ^2} is:

$$S_{\xi^2} = -2\overline{\rho}\widetilde{D}\frac{\partial\widetilde{\xi}}{\partial x_i}\frac{\partial\widetilde{\xi}}{\partial x_i} - 2\overline{\rho}\Omega\left(\widetilde{\xi^2} - \widetilde{\xi}^2\right), \tag{6}$$

$$\Omega = \frac{\widetilde{D}_T + 2\widetilde{D}}{\Delta^2}, \tag{7}$$

where the quantity Ω is known as the mixing frequency [9].

In the flamelet chemistry approach, the material properties $\overline{\rho}, \widetilde{\nu}, \widetilde{D}$ are functions of $\widetilde{\xi}$ and $\widetilde{\xi^2}$ only, the latter two via the resolved temperature, $\widetilde{T}$

$$\overline{\rho} = \overline{\rho}\left(\widetilde{\xi}, \widetilde{\xi^2}\right), \tag{8}$$

$$\widetilde{T} = \widetilde{T}\left(\widetilde{\xi}, \widetilde{\xi^2}\right), \tag{9}$$

$$\widetilde{\nu} = \nu_0\left(\frac{\widetilde{T}}{300K}\right)^{1.69}, \ \frac{\widetilde{\nu}}{\widetilde{D}} = \sigma, \ \sigma = 0.82, \nu_0 = 1.42\times10^{-5}\left[\frac{m^2}{s}\right]. \tag{10}$$

For most turbulent reactive flows with flamelet modeling, the form of the constitutive equations (eqs.8-10) is traditionally determined by performing a laminar opposing jet flame simulation, and assuming that the PDF of the mixture fraction, ξ, belongs to the $\beta-$function family. This then allows us to tabulate every moment of the PDF of ξ - in particular $\overline{\rho}, \widetilde{T}$ – as a function of $\widetilde{\xi}, \widetilde{\xi^2}$. However, in the present work we forgo the flamelet opposed jet solution: instead, we specify the temperature T and specific volume v as quadratic functions of the mixture fraction ξ:

$$v\left(\xi\right) = 7.98 - 23(\xi - 0.551)^2\left[\frac{kg}{m^3}\right], \tag{11}$$

$$T\left(\xi\right) = 2100K - 7200K(\xi - 0.5)^2. \tag{12}$$

As elaborated in Section 5, this is done so that the Favre-averaged values $\widetilde{v}$ and $\widetilde{T}$ are independent of the shape of the PDF of ξ. This allows for consistency between the self-contained solution of eqs. (2-10) (which we shall refer to as a Standalone-LES (S-LES) solution), and the fully-coupled LES/PDF simulation (described in the remainder of this section and in Section 3) with a single scalar for mixture fraction. As seen in Sections 5 and 6, such consistency provides a useful numerical test case for examining the errors inherent to the LES to PDF coupling.

2.2 Governing Equations for the LES component of an LES/PDF simulation

In an LES/PDF simulation, we remove the flamelet modeling from the standalone-LES simulation described in the previous subsection, and replace it with composition PDF modeling of the reaction. Therefore, from all the governing equations of a standalone-LES simulation listed above, the LES component of an LES/PDF simulation solves only eqs.(2,3,7,10). This leaves the evaluation of resolved specific volume and temperature, which is done by the PDF component.

2.3 Governing equations for the PDF component of an LES/PDF simulation

The PDF code takes a Monte Carlo approach to approximating the mass-weighted composition PDF. The simulation domain is discretized into PDF cells, which consist of one or more LES FV cells [18], and, for a specified parameter N_{pc}, each PDF cell contains between $\frac{\sqrt{2}N_{pc}}{2}$ and $\sqrt{2}N_{pc}$ particles. Each particle has a mass m^*, which is unchanged unless that particle is split in two or combined with another (for the purpose of controlling the number of particles in a PDF cell), and a location X_j^* and composition ϕ_α^*. In the equations below, we use the superscript * to denote either an individual particle's property (such as m^* for the mass of the current particle), or the value of an LES field evaluated at that particle's location (such as $\widetilde{\phi}_\alpha^*$ for the resolved mass weighted composition vector at the particle's current location). With this notation, the evolution equations for X_j^* and ϕ_α^* are:

$$dX_j^* = \left[\widetilde{u}_j + \frac{1}{\overline{\rho}}\frac{\partial\left(\overline{\rho}\widetilde{D}_T\right)}{\partial x_j}\right]^* dt + \left[2\tilde{D}_T^*\right]^{1/2} dW_j^*, \tag{13}$$

$$d\phi_\alpha^* = -\Omega^*\left(\phi_\alpha^* - \widetilde{\phi}_\alpha^*\right)dt + \left[\frac{1}{\overline{\rho}}\frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{D}\frac{\partial\widetilde{\phi}_\alpha}{\partial x_j}\right)\right]^* dt + S_\alpha\left(\boldsymbol{\phi}^*\right)dt. \tag{14}$$

In the above equations, Ω denotes the mixing frequency introduced in eq.(7), and $S_\alpha\left(\boldsymbol{\phi}^*\right)$ is the reaction source term. The term dW_j^* denotes a Wiener increment, with the star superscript emphasizing that the Wiener processes for the different particles are independent.

We shall use angled brackets, $\langle\cdot\rangle$, to denote a sum over all particles in a given cell and its immediate neighbors, weighted by a basis function $B\left(X_j^*\right)$ whose support lies within the present cell and its neighbors: for a general particle property g, we have

$$\langle g\rangle = \sum_{cell+neighbors} g^* B\left(X_j^*\right). \tag{15}$$

The two most common examples for $B\left(X_j^*\right)$ are the indicator function of the cell, whose use we shall refer to as the Particle-in-Cell (PIC) approach and a

linearly decreasing tent function which is centered at the cell's center of mass, whose use we shall refer to as the Cloud-in-Cell (CIC) approach [6]. The implementation of CIC is more difficult, since it involves additional communication in order to obtain information about the particles in the neighboring cells. However, CIC comes with the advantage that the continuous form of $B\left(X_j^*\right)$ implies that $\langle g \rangle$ is itself continuous in time, whereas for PIC there is a discontinuous jump in $\langle g \rangle$ as particles enter and leave the cell.

We also apply alternating direction implicit smoothing to the cell mean averages $\langle g \rangle$, and denote the smoothed fields as $\{\langle g \rangle\}$. The smoothing process is described in detail in [6]. Here, we need only note that the amount of smoothing is controlled by a parameter, α, so that implicit smoothing with a given value of α is equivalent to explicit smoothing over α cells in each direction. The value $\alpha = 1.0$ corresponds to no smoothing.

With this definition of a local ensemble mean, $\widetilde{v}, \widetilde{T}$ are defined as

$$\widetilde{v} = \left\{ \frac{\langle mv \rangle}{\langle m \rangle} \right\}, \tag{16}$$

$$\widetilde{T} = \left\{ \frac{\langle mT \rangle}{\langle m \rangle} \right\}, \tag{17}$$

where m^*, v^*, T^* respectively denote a particle's mass, specific volume and temperature. The resolved viscosities and diffusivities are then defined by eqs.(9,10), as in the standalone-LES approach. Whereas in the standalone-LES approach, v, T are functions of the mixture fraction only, in the LES/PDF approach, v, T are functions of the entire composition vector, ϕ_α. As already mentioned, in Sections 5 and 6 we use for our numerical tests a composition vector which consists only of the mixture fraction, for the purposes of comparing standalone-LES and LES/PDF solutions, which allows us to determine the amount of error in the simulation which is due to the LES/PDF coupling. However, it is important to note that in a typical LES/PDF simulation, the composition vector ϕ_α consists of the collection of significant chemical species, with the addition of enthalpy, and so there is no modeling involved in obtaining v and T from ϕ_α. This is one of the advantages of PDF reaction modeling over the flamelet approach. Another advantage is that the source term $S_\alpha(\boldsymbol{\phi})$ on the right hand side of eq.(14) requires no modeling either, provided that the chemical species to be used in the composition vector ϕ_α are appropriately chosen.

3 Coupling between the LES and PDF solutions

In the present LES/PDF algorithm, the LES algorithm uses resolved temperature values obtained from the PDF code, and the PDF code uses values for velocity, molecular and turbulent viscosity obtained from the LES code. Information about specific volume originates in the PDF code, but its transfer to the LES code is challenging, for the following reason: in the solution of the momentum and continuity equations, eqs.(2,3) by the LES solver, pressure is

determined as the solution of a Poisson equation whose source includes the term $\partial\widetilde{v}/\partial t$. Due to the stochastic nature of the PDF solution, however, (eq.13) for a given time step Δt, the statistical error in the increment $\Delta\widetilde{v}$, for a single-particle ensemble, is

$$\varepsilon_{st_1} = C_1 \Delta t^{1/2}, \tag{18}$$

where C_1 is a fixed constant, proportional to $\widetilde{D}_T^{1/2}$. Therefore, the statistical error in the smoothed increment $\Delta\widetilde{v}$, with N_{pc} particles per cell [6], is

$$\varepsilon_{st_2} = \frac{C_2 \Delta t^{1/2}}{\left(N_{pc}\alpha^3\right)^{1/2}}, \tag{19}$$

and so the approximation

$$\partial\widetilde{v}/\partial t \approx \Delta\widetilde{v}/\Delta t \tag{20}$$

contains a statistical error whose magnitude scales in the following manner

$$\varepsilon_{st} = \frac{C}{\left(N_{pc}\alpha^3 \Delta t\right)^{1/2}}, \tag{21}$$

where C is a constant which depends only on the flow geometry and material properties, N_{pc} is the number of particles per cell and α is the smoothing parameter: note that since Δt appears in the denominator of eq.(21), for small values of Δt the error implied by eq.(21) is considerable.

In order to obtain a solution to the Poisson equation for pressure, we can take one of two alternative approaches for specific volume coupling.

3.1 The Particle Specific Volume Approach (PSV)

The PSV approach uses the straightforward procedure of simply passing the PDF values of specific volume to the LES code, and using a value for the smoothing parameter α which is large enough to reduce the error ε_{st} to manageable levels. As we see below, in the numerical tests of coupled LES/PDF turbulent jet simulations, this approach requires the use of smoothing parameter values as large as $\alpha = 4.0$ (which implies that smoothing is performed over 64 cells). Overall, we show below that the PSV approach is not as accurate as the specific volume coupling approach which is described next.

3.2 The Transported Specific Volume Approach (TSV)

The transported specific volume (TSV) approach is based on the fact that the change of resolved specific volume which is due to transport in physical space (eq.13) can be calculated by the LES solver, thus leaving only the specific volume change due to turbulent mixing, molecular diffusion, and chemical reaction (the three terms on the right hand side of eq.14) to be extracted from the PDF code.

In the TSV approach, the LES code solves for an additional scalar: the transported specific volume, $\hat{v}$. This is used as the LES Favre-averaged specific volume. The equation for the transported specific volume, $\hat{v}$, is:

$$\frac{\partial\left(\overline{\rho}\hat{v}\right)}{\partial t}+\frac{\partial\left(\overline{\rho}\widetilde{u}_j\hat{v}\right)}{\partial x_j}=\frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{D}_T\frac{\partial\hat{v}}{\partial x_j}\right)+S_v+\omega_v, \tag{22}$$

where S_v is the specific volume source term due to mixing, molecular diffusion, and chemical reaction, defined by

$$S_v\equiv\left\{\frac{\langle\dot{v}\rangle}{\langle v\rangle}\right\}, \tag{23}$$

where $\dot{v}^*$ is the rate of change of a particle's specific volume due to mixing, molecular diffusion, and chemical reaction. The second term on the right hand side of eq.(22), ω_v, is a relaxation term of the form

$$\omega_v=\overline{\rho}\frac{\widetilde{v}-\hat{v}}{\tau}, \tag{24}$$

with the relaxation time step, τ, set to $\tau=4\Delta t$ in this work. The inclusion of the relaxation term ω_v is necessary to keep the LES specific volume, $\hat{v}$, and the PDF specific volume, $\widetilde{v}$, consistent with each other. In the absence of numerical errors, eq.(22) implies $\hat{v}=\widetilde{v}$, but for a practical reactive flow simulation omitting ω_v from eq.(22) causes $\hat{v}$ and $\widetilde{v}$ to become independent of each other over long time intervals.

Since the LES code is wholly deterministic, there is no statistical error due to transport in the approximation of $\partial\widetilde{v}/\partial t$, and so the value of the constant C on the right hand side of eq.(21) is reduced relative to the PSV implementation of specific volume coupling. This allows the use of smaller values of the smoothing parameter α, which, as we will see in Section 5, yields overall more accurate solutions.

It should be noted that specific volume coupling via a transported scalar equation has previously been used by researchers working on PDF methods for turbulent reactive flows [13, 12, 11, 4]. The contribution of the present work is in the development of a second-order accurate (in both space and time) algorithm for LES/PDF specific volume coupling via a transported scalar equation, in the testing and determination of an optimal specific volume coupling scheme, and in the verification of the overall second-order accuracy of the LES/PDF code with respect to the grid size and time step.

4 Description of a Second-Order Accurate LES/PDF Time Stepping Algorithm

In this section, we describe the coupling procedure between the LES and PDF solvers which yields an overall second-order accurate solution with respect to the grid size and time step, for a fixed LES filter width. This convergence behavior

is demonstrated in Section 6, in which we present results from a numerical test case, which indicate that second-order convergence is indeed achieved. In this section, we describe the LES to PDF coupling procedure that allows second-order convergence of the overall code.

First, we give a short description of the pre-existing time-stepping algorithms for the standalone LES code with flamelet/progress variable chemistry modeling [7], and the particle PDF code with externally specified velocity and diffusivity fields [5]. Then, a description is given to the modifications in the above procedures which yield a fully-coupled second-order accurate LES code with PDF chemistry modeling. For simplicity, we assume that all time steps are of the same length, Δt.

4.1 Time stepping in the standalone-LES code

At the beginning of the time step, we have values for the resolved mixture fraction and its square, resolved density and temperature, $\widetilde{\xi}, \widetilde{\xi^2}, \overline{\rho}$, and $\widetilde{T}$ respectively, at $t = t_0$. From here, we also have resolved viscosity and diffusivity at $t = t_0$, via eqs.(9,10). The velocity, on the other hand, is staggered half a time step back in time: at the beginning of the time step, it is known at $t = t_0 - \Delta t/2$ [7]. The objective of the time step is to obtain the values of $\widetilde{\xi}, \widetilde{\xi^2}, \overline{\rho}$, and $\widetilde{T}$ at $t = t_0 + \Delta t$, and to obtain the values of velocity at $t = t_0 + \Delta t/2$.

In the sub-steps described below, we use the notation $\cdot|^q$ to denote fields at the time level $t = t_0 + q\Delta t$, as in $\widetilde{u}_j|^{-1/2}$ for $\widetilde{u}_j$ at $t = t_0 - \Delta t/2$, and $\overline{\rho}|^0$ for $\overline{\rho}$ at $t = t_0$. In the procedure below, several iterations (whose number is specified by the user, and must be at least two) of sub-steps 2, 3 and 4 are taken for each time step.

1. **Evaluation of turbulent viscosity and diffusivity.** Evaluate the turbulent viscosity and diffusivity values, $\widetilde{\nu}_T, \widetilde{D}_T$ respectively, using the dynamic Smagorinsky procedure on the initial velocity and scalar fields. Set the initial guess for the velocity as $\widetilde{u}_j|^{1/2} = \widetilde{u}_j|^{-1/2}$.

2. **Scalar equations.** If this is the first iteration, set $\widetilde{\xi}\Big|^1 = \widetilde{\xi}\Big|^0$ and $\widetilde{\xi^2}\Big|^1 = \widetilde{\xi^2}\Big|^0$. Using a transport-diffusion solver based on the QUICK scheme, update the increments $\Delta\widetilde{\xi} = \widetilde{\xi}\Big|^1 - \widetilde{\xi}\Big|^0$ and $\Delta\widetilde{\xi^2} = \widetilde{\xi^2}\Big|^1 - \widetilde{\xi^2}\Big|^0$ which are implied by the evolution of eqs.(4,5) forward in time by a time step of length Δt. The velocity used in that time step is $\widetilde{u}_j|^{1/2}$. Increments in a given field are updated by changing the value of that field at the later time level, so that, for example, an update in the increment $\Delta\widetilde{\xi} = \widetilde{\xi}\Big|^1 - \widetilde{\xi}\Big|^0$ implies a change in the field $\widetilde{\xi}\Big|^1$.

3. **Momentum equation.** If this is the first iteration, use the value of $\overline{p}|^0$ from the previous time step, and set $\widetilde{u}_j|^{1/2} = \widetilde{u}_j|^{-1/2}$. Using the

material properties at the beginning of the time step and the current working values for the pressure field at time $t = t_0$, $\overline{p}|^0$, update the velocity increments $\Delta\widetilde{u}_j = \widetilde{u}_j|^{1/2} - \widetilde{u}_j|^{-1/2}$ which are implied by the evolution of the momentum equation, eq.(3) forward in time by a time step of length Δt.

4. **Pressure correction.** Solve the Poisson equation for pressure implied by eqs.(2,3) to update the working pressure, $\overline{p}|^0$. Use the change in $\overline{p}|^0$ to update $\Delta\widetilde{u}_j = \widetilde{u}_j|^{1/2} - \widetilde{u}_j|^{-1/2}$ and ensure that the continuity equation is satisfied.

5. If we are at the last iteration, use the current working values as the end result. If not, go back to sub-step 2.

Note that this description is more narrowly focused on the structure of the LES code's time step than on the numerical solvers used to advance the scalar and momentum equation and to solve for the pressure. For a description of those algorithms, the reader is referred to Pierce and Moin [7].

4.2 Time stepping in the PDF component of an LES/PDF solution

For the LES/PDF algorithm, we adopt one of the weakly second-order accurate splitting schemes for the evolution of the particle positions and composition variables which are described in [5]. In the composition PDF context, splitting schemes are algorithms for evolving the particle equations (eqs.(13,14)) by taking several fractional steps, each of which deals with one of the three physical processes which occur in the evolution of eqs.(13,14): these processes are transport in physical space, mixing and chemical reaction.

The splitting scheme which we use here is referred to as "TCRCT" in [5]. This splitting scheme consists of half a time step of transport in physical space, followed by half a time step of mixing and molecular diffusion, a full time step of reaction, another half time step of mixing and diffusion, and a half time step of transport. We note that, unlike the iterative procedure of the previous subsection, the PDF time stepping requires only one iteration.

Here is a description of the algorithm used to update X_j^*, ϕ_α^* from t_0 to $t_0 + \Delta t$. It is assumed that the velocity, density and diffusivity fields, $\widetilde{u}_j, \overline{\rho}, \widetilde{D}, \widetilde{D}_T$, are known with second-order accuracy at the middle of the time step, $t_0 + \Delta t/2$. Note that the velocity and diffusivity fields are evaluated in the LES component of the LES/PDF solution.

1. **Transport half-step.** Using a weakly second-order accurate SDE integration scheme such as that of Kloeden and Platen [15], advance the particle positions for half a time step, $X_j^*\big|^0 \rightarrow X_j^*\big|^{1/2}$ by taking an increment of length $\Delta t/2$ in eq.(13)

2. **Mixing and diffusion half-step.** Advance the mixing and molecular diffusion processes in time by an increment of length $\Delta t/2$, by taking an increment of length $\Delta t/2$ in the evolution equation

$$d\phi_\alpha^* = -\Omega^* \left(\phi_\alpha^* - \widetilde{\phi}_\alpha^*\right) dt + \left[\frac{1}{\overline{\rho}} \frac{\partial}{\partial x_j} \left(\overline{\rho} \widetilde{D} \frac{\partial \widetilde{\phi}_\alpha}{\partial x_j}\right)\right]^* dt, \tag{25}$$

which is the mixing and molecular diffusion component of the chemical composition evolution equation (eq.14).

3. **Reaction step.** Advance the reaction process in time by an increment of length Δt, by taking an increment of length Δt in the evolution equation

$$d\phi_\alpha^* = S_\alpha \left(\phi^*\right) dt, \tag{26}$$

which is the reaction component of eq.(14).

4. **Mixing and diffusion half-step.** Repeat sub-step 2.

5. **Transport half-step.** Advance the particle positions for half a time step, $X_j^*\big|^{1/2} \rightarrow X_j^*\big|^1$, analogously to sub-step 1.

Sub-steps 2,3,4 use the particle positions $X_j^*\big|^{1/2}$ after the first transport half-step (sub-step 1). Also, all steps 1 through 5 use the values of $\widetilde{u}_j, \overline{\rho}, \widetilde{D}, \widetilde{D}_T$ at the midpoint of the time step: $t = t_0 + \Delta t/2$. The purpose of this particular choice of splitting for the processes of transport, mixing and reaction is that it allows us to take a single reaction time step of length Δt, which reduces overall simulation time due to the fact that the reaction substep (the evolution of eq.(26)) is the most costly component in a PDF simulation.

4.3 Time stepping in the coupled LES/PDF simulation

The LES/PDF coupling scheme proposed here does not make any changes to the PDF time stepping algorithm described in Section 4.2. However, when using the auxiliary scalar approach (TSV) for passing of specific volume information from the PDF to the LES portion of the code, the source term S_v (eq.23) is evaluated by taking the difference in particle specific volumes before and after the mixing, molecular diffusion and reaction substeps. In particular, if v_1^*, v_5^* are respectively the specific volumes of a given particle, as determined by its composition vector ϕ_α, before and after the mixing, molecular diffusion, and reaction substeps (sub-steps 2,3, and 4 in subsection 4.2), then we calculate S_v by

$$S_v = \left\{ \frac{2 \left\langle m^* \left(v_5^* - v_1^*\right)\right\rangle}{\Delta t \left\langle m^* \left(v_5^* + v_1^*\right)\right\rangle} \right\}. \tag{27}$$

Similarly to Section 4.1, at the beginning of the time step we have values for $\overline{\rho}, \widetilde{T}$ at $t = t_0$ and values for $\widetilde{u}_j$ at $t = t_0 - \Delta t/2$. Also, similarly to Section

4.2, at the beginning of the time step we have values for X_j^*, ϕ_α^* at $t = t_0$. In the following algorithm, any steps which are denoted as "TSV only" or "PSV only" are to be skipped if the alternative algorithm for specific volume coupling is used. Also, similarly to the procedure described in subsection 4.1, sub-steps 7 and 8 are iterated a user-specified number of times (at least twice, to achieve overall second-order accuracy in time).

1. **Extrapolation of LES fields forward in time, to the middle of the PDF step.** Evaluate the turbulent viscosity and diffusivity values, $\widetilde{\nu}_T, \widetilde{D}_T$ respectively, using the dynamic Smagorinsky procedure on the initial velocity and scalar fields. Using linear extrapolation on the LES fields $\overline{\rho}, \widetilde{D}, \widetilde{D}_T$, and $\widetilde{u}_j$ from the last two time steps, compute a second-order approximation of the values of $\overline{\rho}, \widetilde{D}, \widetilde{D}_T$, and $\widetilde{u}_j$ at time $t = t_0 + \Delta t/2$. We denote these extrapolated fields as $\left(\overline{\rho}, \widetilde{D}, \widetilde{D}_T, \widetilde{u}_j\right)\Big|^{1/2}$. This extrapolation in time is done in order to provide the PDF algorithm described in the above subsection with the velocity, density and diffusivity fields at the time level necessary for achieving second-order accuracy.

2. **First iteration of auxiliary transported scalar equation (TSV only).** Using $\left(\overline{\rho}, \widetilde{D}, \widetilde{D}_T, \widetilde{u}_j\right)\Big|^{1/2}$ and the initial values for $S_v, \widetilde{v}$, respectively $S_v|^{-1/2}, \widetilde{v}|^0$, update the increment $\Delta\hat{v} = \hat{v}\,|^1 - \hat{v}\,|^0$ implied by the evolution of eq.(22) forward in time by a time step of length Δt. The same transport-diffusion solver is used as in sub-step 2 of the standalone LES solution.

3. **PDF time step.** Using the extrapolated fields $\left(\overline{\rho}, \widetilde{D}, \widetilde{D}_T, \widetilde{u}_j\right)\Big|^{1/2}$ obtained from the LES solver, perform the PDF time step described in Section 4.2. Calculate $S_v\,|^{1/2}$ via eq.(27), and calculate $\widetilde{v}\,|^1, \widetilde{T}\,|^1$ from the particle ensemble after the PDF time step, via eqs.(16,17).

4. **Second, and final, iteration of auxiliary transported scalar equation (TSV only).** Using $\left(\overline{\rho}, \widetilde{D}, \widetilde{D}_T, \widetilde{u}_j\right)\Big|^{1/2}$ and the updated values for $S_v, \widetilde{v}$ at $t = t_0 + \Delta t/2$, respectively $S_v\,|^{1/2}$ and $\left(\widetilde{v}\,|^1 + \widetilde{v}\,|^0\right)\Big/ 2$, update the increment $\Delta\hat{v} = \hat{v}\,|^1 - \hat{v}\,|^0$ implied by the evolution of eq.(22) forward in time by a time step of length Δt. This second iteration is performed so that the solution for the transported scalar is second-order accurate in time.

5. **Evaluation of LES resolved specific volume.** For PSV, set $\overline{\rho}\,|^1 = (1/\widetilde{v})|^1$. For TSV, set $\overline{\rho}\,|^1 = (1/\hat{v})|^1$.

6. **Evaluation of molecular density and diffusivity at the end of the PDF step.** Using $\widetilde{v}\,|^1, \widetilde{T}\,|^1$, calculate $\widetilde{D}\,|^1, \widetilde{\nu}\,|^1$. This information is not used until the next time step.

7. **Iteration of momentum equation.** If this is the first iteration of sub-steps 7 and 8, use the value of $\overline{p}|^0$ from the previous time step, and use the value of $\widetilde{u}_j|^{1/2}$ obtained at sub-step 1. Using the initial transport properties, $\widetilde{\nu}|^0, \widetilde{T}|^0, \widetilde{D}|^0, \widetilde{\nu}|^0$, $\widetilde{D}_T|^0, \widetilde{\nu}_T|^0$, and the current working values for the pressure field, $\overline{p}|^0$, update the velocity increments $\Delta\widetilde{u}_j{=}\widetilde{u}_j|^{1/2} - \widetilde{u}_j|^{-1/2}$ which are implied by the evolution of the momentum equation, eq.(3), forward in time by a time step of length Δt.

8. **Iteration of the pressure correction.** Solve the Poisson equation for pressure implied by eqs. (1,2) to update the working pressure, $\overline{p}|^0$. Use the change in $\overline{p}|^0$ to update $\Delta\widetilde{u}_j{=}\widetilde{u}_j|^{1/2} - \widetilde{u}_j|^{-1/2}$ and ensure that the continuity equation is satisfied.

9. If we are at the last iteration of sub-steps 7 and 8, use the current working values $\widetilde{u}_j|^{1/2}, \overline{p}|^0$, as the end result for velocity and pressure. If not, go back to sub-step 7.

Note that, unlike the standalone-LES simulation, which requires iteration of the scalar transport-diffusion solver, the above algorithm requires only one PDF time step for each LES/PDF time step. This is intentional, as the cost of a PDF time step, for a typical PDF simulation with at least 20 particles per cell, is much greater than the cost of an LES time step.

In sections 5 and 6, we test the performance of this LES/PDF coupling algorithm.

5 Numerical Testing of Alternative Coupling Strategies: Turbulent Jet Bluff-Body Flame

In this section, we compare the performance of the alternative choices for LES to PDF coupling schemes, in order to establish which provides optimal performance for a turbulent test flow representative of modern applications of LES/PDF methods.

Firstly, we establish a criterion for measuring the performance of the coupling scheme, apart from that of other aspects of the LES/PDF code. In order to do this, we specify a chemical model and material properties which can be solved consistently by both a standalone-LES (S-LES) simulation and a fully-coupled LES/PDF solution. Then, the coupling error is defined as the difference between the S-LES and the LES/PDF solutions.

To this end, we use the flamelet model without progress variable as described in Section 2. Then, in order to model subfilter variance, a standalone-LES simulation solves for the resolved mixture fraction, $\widetilde{\xi}$, and the resolved square mixture fraction, $\widetilde{\xi^2}$, and estimates resolved quantities by assuming that the shape of the PDF of the mixture fraction, $f(\xi)$, belongs to the $\beta-$function family. On the other hand, an LES/PDF simulation approximates the exact functional form of $f(\xi)$ without assumptions. Therefore, the shapes of the PDF

of mixture fraction yielded by the two alternative solution methods are bound to differ: in order to account for this, we set the relevant material properties – specific volume and temperature – to vary quadratically with mixture fraction, as formulated in eqs.(11),(12). The values of molecular viscosity and diffusivity are defined by eq.(10).

This quadratic variation of v and T implies that their Favre mean is a function only of the Favre mean and variance of ξ, $\widetilde{\xi}$ and $\widetilde{\xi''^2}$, respectively, and does not depend on the shape of $f(\xi)$. Since $\widetilde{\xi''^2} = \widetilde{\xi^2} - \left(\widetilde{\xi}\right)^2$, this implies that $\widetilde{v}, \widetilde{T}$ are functions of $\widetilde{\xi}, \widetilde{\xi^2}$ only, and hence the governing equations yielded by the standalone-LES and coupled LES/PDF methodologies are consistent.

This specification of material properties is applied to the geometry of the Sandia/Sydney Bluff Body Flame, flame HM1 as first described by Masri and Bilger [14]. In this canonical test flame, a jet of diameter $3.6mm$ and bulk velocity of $118m/s$ is located inside a bluff body of diameter $50mm$, surrounded by a fast coflow whose velocity is $40m/s$. The Reynolds number based on the jet velocity and radius is 14950, and in our simulations the turbulence is modeled by the dynamic Smagorinsky model. Denoting the radius of the bluff body as R_B, in the present simulations we use a computational domain which has the following extent: $x \in [0, 10R_B]$, $r \in [0, 3R_B]$, where x denotes distance downstream from the jet, and r is the radial distance from the jet centerline. The domain is discretized on a uniform cylindrical grid of size $128 \times 128 \times 64$ in the axial, radial and azimuthal directions, respectively, and the nominal number of particles per cell is $N_{pc} = 50$.

Here, we emphasize that we do not aim to simulate the HM1 flame accurately, but rather to use its features in order to ensure that we measure the coupling error between S-LES and LES/PDF simulations in a flow which resembles typical applications of the LES/PDF methodology. It is for this reason that we can use the small computational domain and simple chemistry described above.

We consider three different coupling implementations. The first uses the transported specific volume approach (TSV) with cloud-in-cell (CIC) mean estimation; the second uses TSV with particle in cell (PIC) mean estimation, and the third uses the particle specific volume (PSV) with CIC mean estimation. For each of these implementations, we test different values for the smoothing parameter from the set $\alpha \in \{1.0, 2.0, 3.0, 4.0, 6.0\}$. For each of the 15 alternative simulations outlined above, the LES/PDF algorithm is run for 100 flow-through times, based on the coflow velocity. In all cases, the solution has become statistically stationary by the 30^{th} flow-through time. Statistics are calculated over the latter half of the simulation, after the 50^{th} flow though time.

Here, we consider as statistics the fields of the mean resolved axial velocity, $mean\left(\widetilde{u}_1\right)(\mathbf{x})$, the variance of the resolved axial velocity $var\left(\widetilde{u}_1\right)(\mathbf{x})$, the mean resolved density, $mean\left(\overline{\rho}\right)(\mathbf{x})$, and the variance of the resolved density, $var\left(\overline{\rho}\right)(\mathbf{x})$. Means and variances are computed by averaging over the simulation's time interval and the azimuthal direction, θ. In order to determine

the optimal value of the smoothing parameter α for each coupling scheme, we choose that value of the parameter which minimizes the L^1 differences between $mean\left(\widetilde{u}_1\right), var\left(\widetilde{u}_1\right), mean\left(\overline{\rho}\right)$ and $var\left(\overline{\rho}\right)$ yielded by the LES/PDF algorithm and those yielded by the S-LES solution.

More specifically, the L^1 error definition based on mean resolved axial velocity is as follows

$$\varepsilon_{mean(\widetilde{u}_1)} = \frac{\int |mean(\widetilde{u}_1)_{LES/PDF}\left(\mathbf{x}\right) - mean(\widetilde{u}_1)_{S-LES}\left(\mathbf{x}\right)||\mathrm{d}\mathbf{x}|}{\int |mean(\widetilde{u}_1)_{S-LES}\left(\mathbf{x}\right)||\mathrm{d}\mathbf{x}|}. \tag{28}$$

The errors for $var\left(\widetilde{u}_1\right), mean\left(\overline{\rho}\right)$ and $var\left(\overline{\rho}\right)$, respectively $\varepsilon_{var(\widetilde{u}_1)}, \varepsilon_{mean(\overline{\rho})}$ and $\varepsilon_{var(\overline{\rho})}$, are analogously defined. Tables 1, 2, and 3 present the values for $\varepsilon_{mean(\widetilde{u}_1)}, \varepsilon_{var(\widetilde{u}_1)}, \varepsilon_{mean(\overline{\rho})}$ and $\varepsilon_{var(\overline{\rho})}$ for all simulation cases. Based on these results, we conclude that for PSV/CIC, the optimal choice for smoothing parameter is $\alpha = 4.0$, for TSV/PIC it is $\alpha = 2.0$, and for TSV/CIC it is $\alpha = 1.0$.

Error Type	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$	$\alpha = 4.0$	$\alpha = 6.0$
$\varepsilon_{mean(\widetilde{u}_1)}$	N/A	2.5×10^{-2}	1.7×10^{-2}	$\mathbf{9.7\times10^{-3}}$	2.2×10^{-2}
$\varepsilon_{var(\widetilde{u}_1)}$	N/A	1.9×10^{-1}	8.9×10^{-2}	$\mathbf{4.5\times10^{-2}}$	1.7×10^{-1}
$\varepsilon_{mean(\overline{\rho})}$	N/A	3.7×10^{-3}	2.3×10^{-3}	$\mathbf{2.2\times10^{-3}}$	3.5×10^{-3}
$\varepsilon_{var(\overline{\rho})}$	N/A	3.5×10^{-2}	1.7×10^{-2}	$\mathbf{1.4\times10^{-2}}$	2.7×10^{-2}

Table 1: L^1 errors for PSV/CIC simulations. Minimal error values are shown in bold type. There is no data point for $\alpha = 1.0$: at that value for the smoothing parameter, the PSV/CIC code is unstable.

Error Type	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$	$\alpha = 4.0$	$\alpha = 6.0$
$\varepsilon_{mean(\widetilde{u}_1)}$	6.7×10^{-3}	$\mathbf{5.0\times10^{-3}}$	8.0×10^{-3}	1.0×10^{-2}	1.7×10^{-2}
$\varepsilon_{var(\widetilde{u}_1)}$	2.2×10^{-2}	$\mathbf{1.7\times10^{-2}}$	2.4×10^{-2}	2.8×10^{-2}	3.3×10^{-2}
$\varepsilon_{mean(\overline{\rho})}$	2.1×10^{-3}	$\mathbf{1.5\times10^{-3}}$	2.4×10^{-3}	2.9×10^{-3}	4.2×10^{-3}
$\varepsilon_{var(\overline{\rho})}$	1.0×10^{-2}	$\mathbf{8.4\times10^{-3}}$	1.2×10^{-2}	1.4×10^{-2}	1.5×10^{-2}

Table 2: L^1 errors for TSV/PIC simulations. Minimal error values are shown in bold type.

Error Type	$\alpha = 1.0$	$\alpha = 2.0$	$\alpha = 3.0$	$\alpha = 4.0$	$\alpha = 6.0$
$\varepsilon_{mean(\widetilde{u}_1)}$	$\mathbf{4.7 \times 10^{-3}}$	5.7×10^{-3}	6.5×10^{-3}	8.1×10^{-3}	1.5×10^{-2}
$\varepsilon_{var(\widetilde{u}_1)}$	$\mathbf{9.8 \times 10^{-3}}$	1.2×10^{-2}	1.4×10^{-2}	1.6×10^{-2}	2.0×10^{-2}
$\varepsilon_{mean(\overline{\rho})}$	1.1×10^{-3}	$\mathbf{1.0 \times 10^{-3}}$	1.3×10^{-3}	1.8×10^{-3}	2.7×10^{-3}
$\varepsilon_{var(\overline{\rho})}$	$\mathbf{6.3 \times 10^{-3}}$	6.6×10^{-3}	8.0×10^{-3}	9.7×10^{-3}	1.1×10^{-2}

Table 3: L^1 errors for TSV/CIC simulations. Minimal error values are shown in bold type.

Next, we compare the three solutions with optimal smoothing parameter values for the respective coupling scheme, in order to arrive at the optimal coupling scheme overall. Radial profiles of the statistics are examined at three axial locations: at $x/R_B = 0.27, 2.18$ and 3.96. Figure 1 presents a comparison between the mean resolved axial velocity profiles of the three LES/PDF solutions and the S-LES solution. As we can see, both the TSV/PIC and the TSV/CIC implementation agree well with the S-LES solution. On the other hand, the PSV/CIC solution underpredicts the S-LES mean resolved axial velocity close the centerline, for the intermediate location of $x/R_B = 2.18$. The variance of the resolved axial velocity is more sensitive to the coupling scheme – its profiles are plotted on fig.2. Here, all three schemes are in good agreement with the S-LES solution near the inlet, at $x/R_B = 0.27$; at the intermediate location of $x/R_B = 2.18$, the two TSV solutions fall close to the S-LES solution, whereas the PSV/CIC solution overpredicts the peak value by 20%. The results are most sensitive at the downstream location of $x/R_B = 3.96$, where the TSV/CIC, TSV/PIC and PSV/CIC solutions overpredict the peak variance of the S-LES solution by respectively 11%, 43% and 314%.

Next, we examine profiles for mean resolved density and variance of the resolved density. Figure 3 shows the results for mean resolved density: it can be seen that the results are not sensitive to the coupling scheme – at $x/R_B = 0.27,\ 2.18$ all three LES/PDF solutions are in good agreement with the S-LES solution, whereas at $x/R_B = 3.96$ the TSV/CIC implementation falls almost on top of the S-LES curve, whereas the other two solutions: TSV/PIC and PSV/CIC, underpredict it slightly. Finally, profiles of the variance of the resolved density can be seen on fig.4. Here, all three LES/PDF solutions underpredict the S-LES solution at $x/R_B = 0.27,\ 2.18$, whereas the TSV/PIC solution overpredicts slightly at $x/R_B = 3.96$. Overall, the TSV/CIC scheme is in best agreement with the S-LES results.

From the examination of the radial profiles of velocity and density statistics of the most accurate TSV/CIC, TSV/PIC and PSV/CIC solutions, we conclude that the optimal coupling scheme is TSV/CIC with smoothing parameter $\alpha = 1.0$ (which corresponds to no smoothing). Moreover, we note that this LES/PDF solution can accurately match the S-LES profiles, even for the most

sensitive statistic – variance of axial velocity. Since the present test case is hydrodynamically similar to the Sandia/Sydney Bluff Body Flame, which is a representative application for an LES/PDF solution, this leads us to the conclusion that TSV/CIC is a satisfactory coupling scheme for LES/PDF methods.

6 Numerical Test for Verification of Second-Order Convergence: Smooth Vortex Ring Flow

In the previous section, we established that for a representative turbulent flow the optimal LES/PDF coupling strategy is TSV/CIC coupling with no smoothing ($\alpha = 1.0$). Here, we test, for this coupling implementation, the overall second-order convergence behavior of the code, as the cell size and time step are decreased, and the number of particles per cell is increased.

The computational domain is a cylinder of axial length $2.0m$ and radius $1.5m$. Using $\chi_E(q)$ to denote the indicator function of $q \in E$, for a given set E, the initial velocity field is axi-symmetric, analytically specified as a superposition of a vortex ring and a jet (the following equations are in MKS units, which are omitted in order to avoid clutter):

$$\overline{\rho}\widetilde{u}_x\left(x,r,t=0\right) = 0.5 \times e^{-r^2/4} - \chi_1 sin^2\left(2\pi\left(q_1 - 0.05\right)\right) \times \left((r-0.65)/(rq_1)\right), \tag{29}$$

$$\overline{\rho}\widetilde{u}_r\left(x,r,t=0\right) = \chi_1 sin^2\left(2\pi\left(q_1 - 0.05\right)\right) \times \left((x-0.65)/(rq_1)\right), \tag{30}$$

$$\overline{\rho}\widetilde{u}_\theta\left(x,r,t=0\right) = 0, \tag{31}$$

$$q_1 = \left((x-0.65)^2 + (r-0.65)^2\right)^{1/2}, \tag{32}$$

$$\chi_1 = \chi_{[0.05,0.55]}\left(q_1\right). \tag{33}$$

Similarly, the initial condition for the mixture fraction mean and variance has the following analytic form:

$$\widetilde{\xi}\left(x,r,t=0\right) = 0.1 + 0.8 \times \chi_2 cos^2\left(\pi q_2\right), \tag{34}$$

$$q_2 = \left((x-0.65)^2 + (max(r-0.5,0) \times 3/5)^2\right)^{1/2}, \tag{35}$$

$$\chi_2 = \chi_{[0,0.5]}\left(q_2\right), \tag{36}$$

$$\xi^{''}\left(x,r,t=0\right) = 0.5\widetilde{\xi}\left(1-\widetilde{\xi}\right). \tag{37}$$

The material property definitions are analogous to the previous test case, with the exception that the molecular viscosity and diffusivity, ν and D, have been scaled by a constant ($\nu_0 = 1.5 \times 10^{-3}\left[\frac{m^2}{s}\right]$ in eq.(10)) in order to yield a value for the Reynolds number of $Re = 3000$, based on the cylinder's radius and the maximal velocity in the initial condition. For turbulence modeling we use a large filter of fixed size $\Delta = 0.5m$, instead of the traditional dynamic Smagorinsky

procedure. This yields a grid-independent solution, which is necessary in order to test the overall order of convergence of the LES/PDF code.

The order of convergence is tested by taking the differences between the final values of $\widetilde{u}_1, \widetilde{\xi}, \widetilde{\xi^2}$, and $\overline{\rho}$, as obtained by the LES/PDF simulation, and a highly resolved S-LES solution on a $128 \times 128 \times 64$ grid. These differences are averaged over the azimuthal direction, integrated against a collection of 16 Fourier modes in $x - r$ space, and the overall error is defined as the root-mean-square error of these 16 functionals.

More concretely, for measuring error based on the resolved axial velocity, we define the functionals $g_{j,k}$ by

$$g_{j,k} = \int_{x\in[0,2], r\in[0,1.5]} \widetilde{u}_1(x, r, \theta)\, e^{i\pi(2jx+1.5kr)} r dr dx d\theta, \tag{38}$$

and then we define the error measure for resolved axial velocity, $\varepsilon_{\widetilde{u}_1}$, as

$$\varepsilon_{\widetilde{u}_1} = \left(\sum_{j,k=1}^{4} \left| E\left(g_{j,k}^{LES/PDF} - g_{j,k}^{S-LES} \right) \right|^2 \right)^{1/2}, \tag{39}$$

where we use $E(\cdot)$ to denote the expectation of a random variable. The error measures for resolved mixture fraction, resolved square mixture fraction, and resolved density, $\varepsilon_{\widetilde{\xi}}, \varepsilon_{\widetilde{\xi^2}}$ and $\varepsilon_{\overline{\rho}}$ respectively, are similarly defined. Convergence of the LES/PDF algorithm with respect to these error measures verifies the weak convergence properties of the method. In the present context, weak convergence is taken to mean convergence of the expectations of general linear functionals of the end solution, as opposed to standard pointwise convergence.

Convergence studies of LES/PDF computational algorithms, such as the one we present here, are rare [5,6], due to the high cost introduced by the stochastic nature of the PDF aspect of the code. In particular, for a grid with a cell size of Δx, a time step of length Δt, N_{pc} particles per cell and N_p particles total, and a fixed value of the smoothing parameter α [6], second-order convergence of the overall code with respect to the grid size and time step implies that the errors in the functionals $g_{j,k}$ scale in the following manner:

$$\varepsilon = C_1 \Delta x^2 + C_2 \Delta t^2 + C_3 \frac{1}{N_{pc}} + C_4 \left(\frac{1}{N_p} \right)^{1/2} Y. \tag{40}$$

In the above equation, the four terms on the right represent, respectively, errors due to grid resolution, time step, statistical bias and statistical errors, where Y is a Gaussian random variable of zero mean and unit variance. The last component in the above expression, which is due to statistical error, illustrates the advantage of using error norms based on linear functional of the solution, as opposed to pointwise error estimates: for the latter, the statistical error scales as $\left(\frac{1}{N_{pc}}\right)^{1/2}$, which is much larger than $\left(\frac{1}{N_p}\right)^{1/2}$.

Even for the linear functionals considered above, the bias error scales as $\frac{1}{N_{pc}}$. The scaling of the bias error implies that, in order to test second-order

convergence with respect to the grid and time step, we need to increase the number of particles per cell by a factor of 4 each time that Δx and Δt are decreased by a factor of 2, and hence the overall number of particles is increased by a factor of 32.

We perform simulations on five successively more refined grids. The simulation parameters are summarized in the table below

Simulation Type	Grid size ($n_x \times n_r \times n_\theta$)	Particles per cell (N_{pc})	Time step, Δt
S1	$16 \times 16 \times 8$	20	0.0160
S2	$24 \times 24 \times 8$	35	0.0106
S3	$32 \times 32 \times 16$	50	0.0080
S4	$48 \times 48 \times 16$	112	0.0066
S5	$64 \times 64 \times 32$	200	0.0040

In the above table, note that, for N_{pc} to be proportional to Δx^{-2}, N_{pc} for $S1$ would have to be 12 (or 13), and N_{pc} for $S2$ would have to be 28. The higher numbers of $N_{pc} = 20$ for $S1$ and $N_{pc} = 35$ for $S2$ are used in order to ensure a stable run of the particle PDF code. For the simulations $S3, S4, S5$, on whose data points the second-order convergence is primarily based, the relationship $N_{pc} \propto \Delta x^{-2}$ is maintained. Contours of the resolved axial velocity and resolved mixture fraction at the end time, $t = 0.45$, are shown on fig.5.

For each of the simulation types $S1$ through $S5$, we perform, for the purpose of estimating confidence intervals for $g_{j,k}$, 8 independent simulations, with different initial seeds for the random number generator. The 95% confidence interval width for the error measures $\varepsilon_{\widetilde{u}_1}, \varepsilon_{\widetilde{\xi}}, \varepsilon_{\widetilde{\xi^2}}, \varepsilon_{\overline{\rho}}$ is estimated by the formula:

$$CI\ width = 1.96 \times \sqrt{\frac{1}{8} \sum_{j,k=1}^{4} Var\left(g_{j,k}\right)}. \tag{41}$$

The computed error from these simulations can be seen on fig.6, which plots, on a log-log scale, the means and confidence intervals for $\varepsilon_{\widetilde{u}_1}, \varepsilon_{\widetilde{\xi}}, \varepsilon_{\widetilde{\xi^2}}, \varepsilon_{\overline{\rho}}$ against Δx, the grid cell size in the axial direction. Since Δt is directly proportional to Δx for $S1$ through $S5$, second order convergence with respect to the grid and time step corresponds to the data points falling on a straight line of slope 2 in this log-log plot.

As can be seen on fig.6, all four error measures indicate second-order convergence – the reference line of slope 2 passes through the confidence intervals of the $S3, S4, S5$ data points, and with the exception of the $S1$ data points for $\varepsilon_{\widetilde{\xi^2}}$ and $\varepsilon_{\widetilde{u}_1}$, the errors for the coarse-grid $S1$ and $S2$ simulation types are also close to the reference line of slope 2. From these results, we conclude that the LES/PDF scheme implemented in this work is indeed second-order accurate with respect to the grid size and time step.

7 Conclusions

In this paper, we have addressed the issue of coupling between the LES and PDF components of an LES/PDF algorithm for turbulent combustion simulations. A coupling methodology has been proposed which allows for second-order overall accuracy of the algorithm with respect to the grid cell size and the time step. Using a numerical test case based on the turbulent Sandia/Sydney Bluff Body Flame, it has been determined that the optimal coupling scheme is that which uses the auxiliary transported specific volume approach with cloud in cell mean estimation and no smoothing. Finally, for this choice of coupling scheme, convergence studies have been performed to verify the second-order accuracy of the LES/PDF algorithm.

Acknowledgements

This work is supported in part by the Air Force Office of Scientific Research, Grant FA 9550-09-1-0047, and by NASA Grant NNX08A B 36A.

References

[1] S. B. Pope, *PDF Methods for Turbulent Reactive Flows*, Prog. Energy Combust. Sci., **11**, pp. 119-192 (1985)

[2] S. B. Pope, *Turbulent Flows*, Cambridge University Press, Cambridge, 2000

[3] D. C. Haworth, *Progress in Probability Density Function Methods for Turbulent Reacting Flows*, Prog. Energy Comb. Sci., **36**, 2, (2010), pp. 168-259

[4] Y. Z. Zhang, and D. C. Haworth, *A General Mass Consistency Algorithm for Hybrid Particle/Finite Volume PDF Methods*, J. Comp. Phys., **194**, (2004), pp. 156-193

[5] H. Wang, P. P. Popov, and S. B. Pope, *Weak Second-Order Splitting Schemes for Lagrangian Monte Carlo Particle Methods for the Composition PDF/FDF Transport Equations*, J. Comput. Phys. **229**, (2010), pp. 1852-1878

[6] S. Viswanathan, H. Wang, and S. B. Pope, *Numerical Implementation of Mixing and Molecular Transport in LES/PDF Studies of Turbulent Reacting Flows*, J. Comput. Phys. **230**, (2011), pp. 6916-6957

[7] C. D. Pierce, and P. Moin, *Progress-Variable Approach for Large-Eddy Simulation of Turbulent Combustion*, J. Fluid Mech.**504**, (2004), pp. 73-97

[8] V. A. Sabel'nikov, and O. Soulard, *Rapidly Decorrelating Velocity-Field Model as a Tool for Solving One-Point Fokker-Planck Equations for Probability Density Functions of Turbulent Reactive Scalars*, Phys. Rev. E, **72** (2005)

[9] P. Colucci, F. Jaberi, P. Givi, and S. B. Pope, *Filtered Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, Phys. Fluids **10** (1998), pp.499-515

[10] F. Jaberi, P. Colucci, S. James, P. Givi, and S. B. Pope, *Filtered Mass Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, J. Fluid Mech. **401** (1999), pp.85-121

[11] V. Raman, and H. Pitsch, *A Consistent LES/Filtered Density Function Formulation for the Simulation of Turbulent Flames with Detailed Chemistry*, Proc. Comb. Inst., **31**, (2007), pp. 1711-1719

[12] Y. Ge, M. J. Cleary, and A. Y. Klimenko, *Sparse-Lagrangian FDF Simulations of Sandia Flame E with Density Coupling*, Proc. Comb. Inst., **33**, (2011)

[13] M. Muradoglu, P. Jenny, S. B. Pope, and D. A. Caughey, *A Consistent Hybrid Finite-Volume/Particle Method for the PDF Equations of Turbulent Reactive Flows*, J. Comput. Phys., **154** (1999), pp. 342-371

[14] A. R. Masri, and R. W. Bilger, *Turbulent Diffusion Flames of Hydrocarbon Fuels Stabilized on a Bluff Body*, Proc. Comb. Inst., **20**, (1985)

[15] P. E. Kloeden, and E. Platen *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, Berlin, 1992

[16] S. B. Pope *Computations of Turbulent Combustion: Progress and Challenges*, Proc. Comb. Inst., **23**, (1991), pp. 591-612

[17] H. Pitsch *Large-Eddy Simulation of Turbulent Combustion*, Annu. Rev. Fluid Mech., **38**, (2006), pp. 453-482

[18] H. Wang, and S. B. Pope *Large Eddy Simulation/Probability Density Function Modeling of a Turbulent CH4/H2/N2 Jet Flame*, Proc. Comb. Inst., **33**, (2011), pp. 1319-1330

[19] K. A. Kemenov, H. Wang, and S. B. Pope *Turbulence Resolution Scale Dependence in Large-Eddy Simulations of a Jet Flame*, Flow Turb. Comb. (2011), DOI: 10.1007/s10494-011-9380-x.

[20] R. Mustata, L. Valino, C. Jimenez, W. P. Jones, S. Bondi *A Probability Density Function Eulerian Monte Carlo Field Method for Large Eddy Simulations: Applications to a Turbulent Piloted Methane/Air Diffusion Flame (Sandia D)*, Combust. Flame, **145**, (2006), pp. 88-104

[21] L. Valiño *A Field Monte Carlo Formulation for Calculating the Probability Density Function of a Single Scalar in a Turbulent Flow*, Flow Turb. Combust., **60**, (1998), pp. 157-172

[22] S. B. Pope *A Monte Carlo Method for the PDF Equations of Turbulent Reactive Flow*, Combust. Sci. Tech., **25**, (1981), pp. 159-174

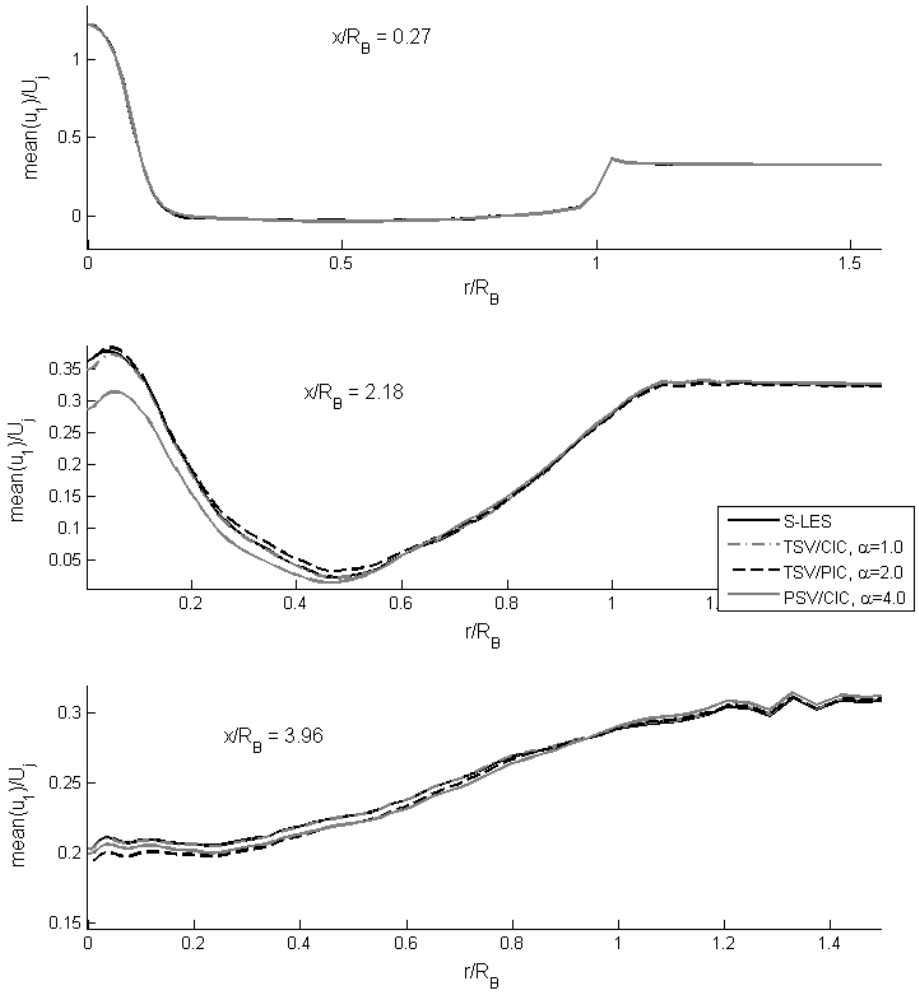

Figure 1: Comparison of the radial profiles of the mean of the resolved axial velocity yielded by the LES/PDF coupling schemes with optimal values for the smoothing parameter α

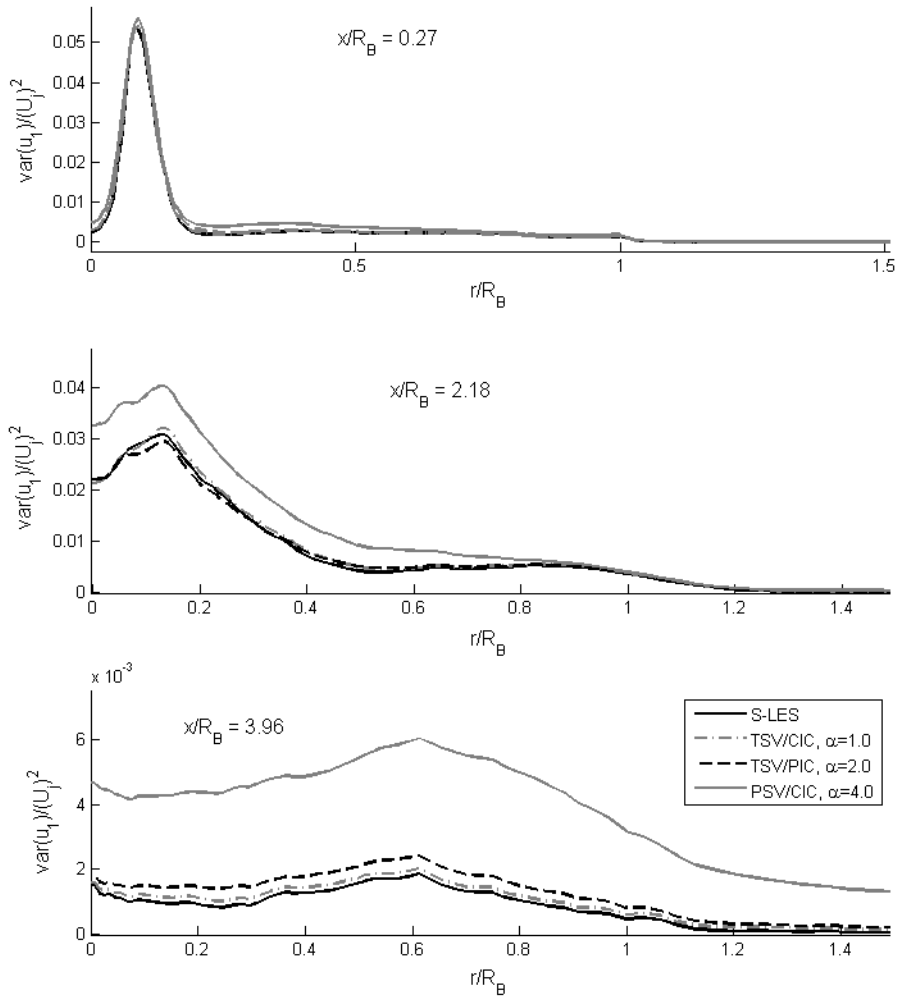

Figure 2: Comparison of the radial profiles of the variance of the resolved axial velocity yielded by the LES/PDF coupling schemes with optimal values for the smoothing parameter α

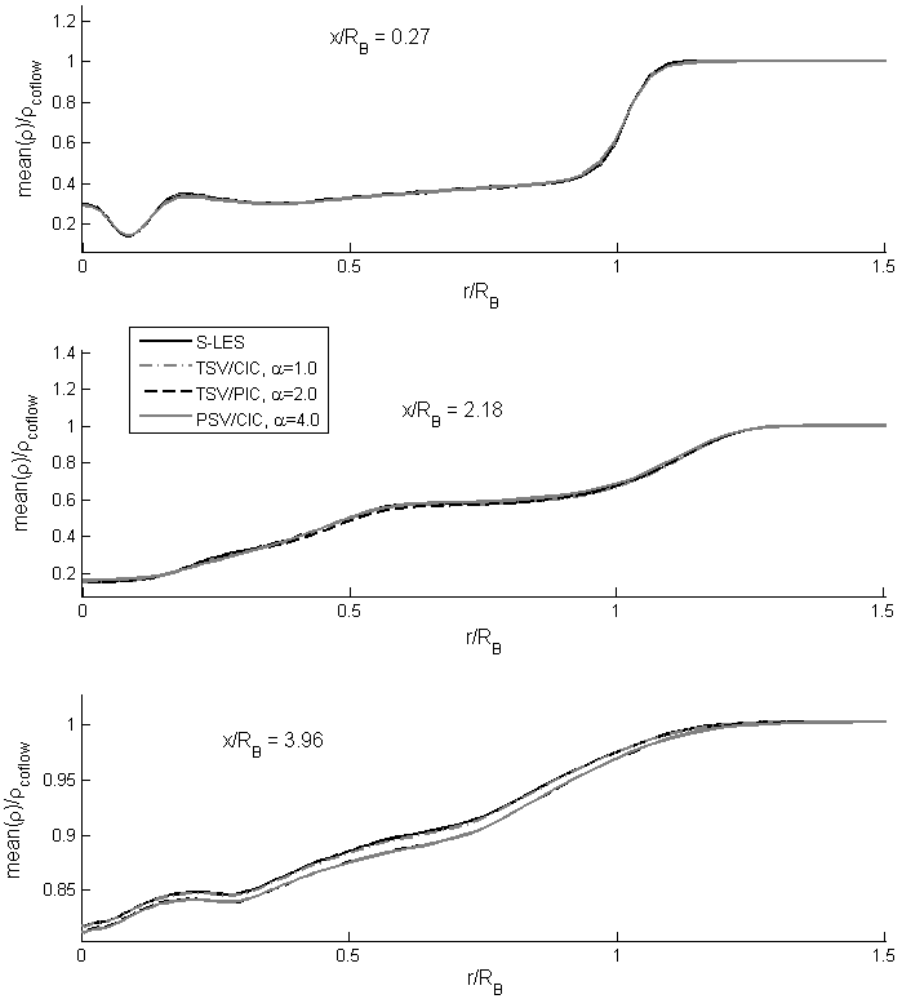

Figure 3: Comparison of the radial profiles of the mean of the resolved density yielded by the LES/PDF coupling schemes with optimal values for the smoothing parameter α

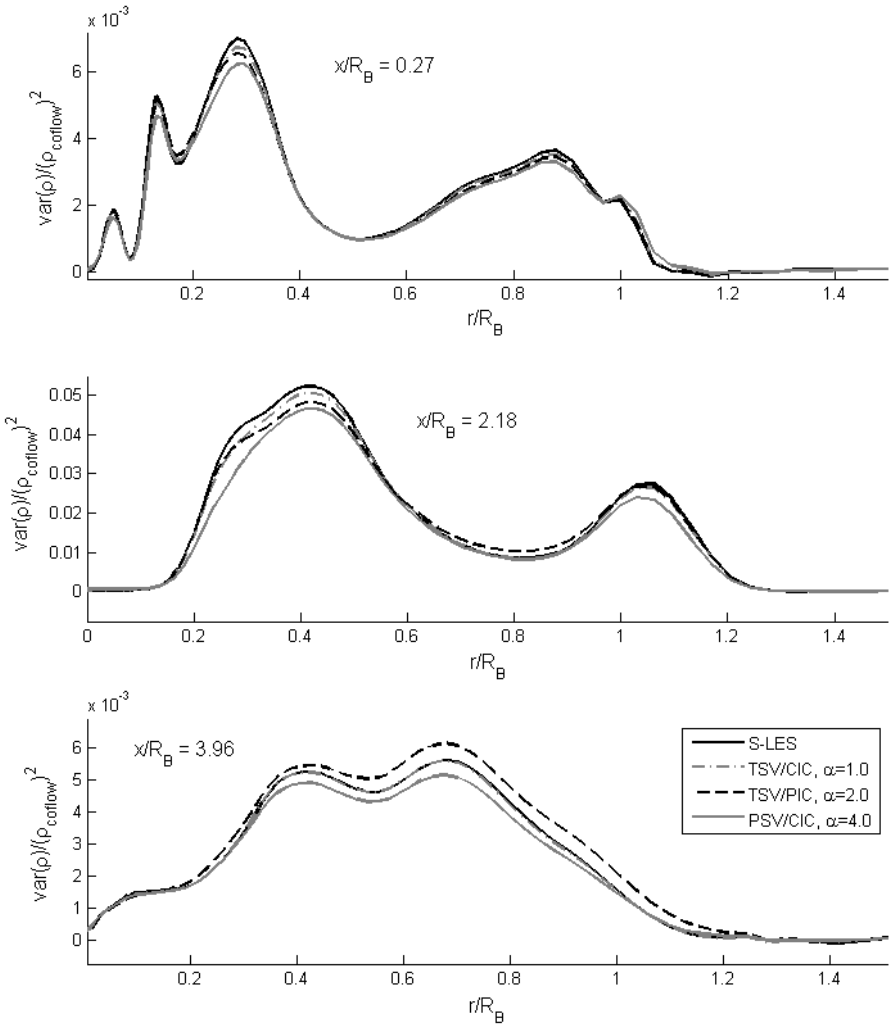

Figure 4: Comparison of the radial profiles of the variance of the resolved density yielded by the LES/PDF coupling schemes with optimal values for the smoothing parameter α

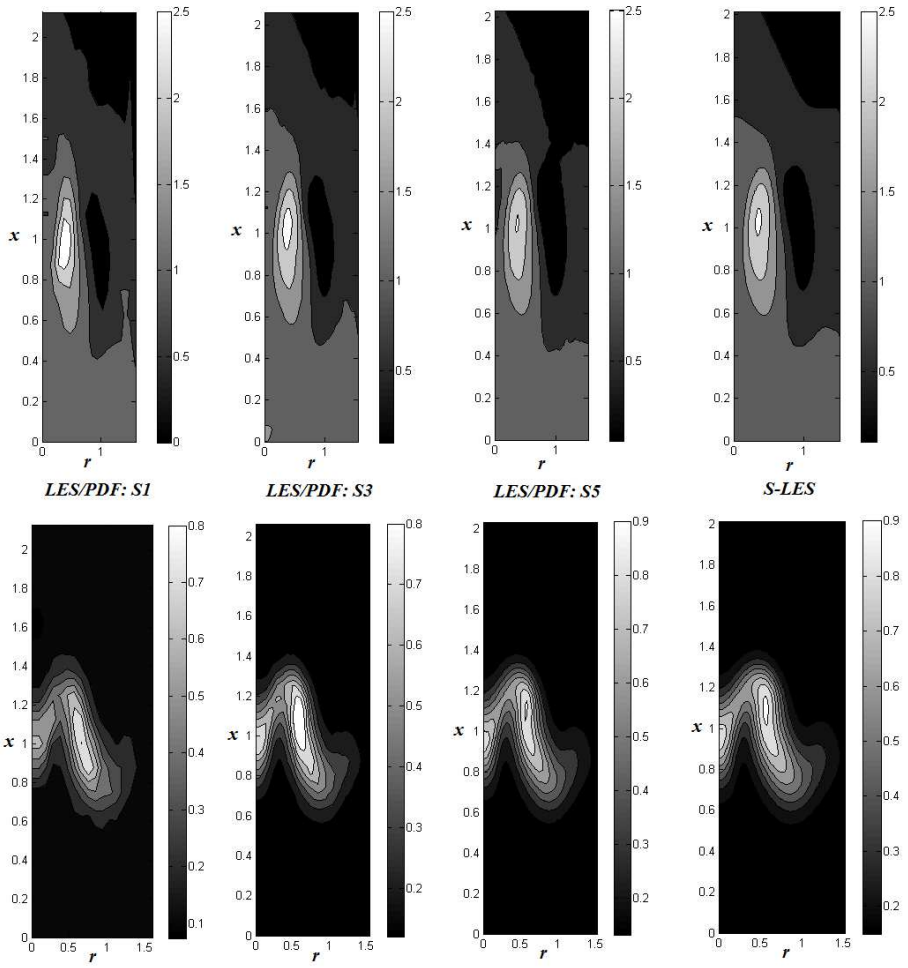

Figure 5: Resolved axial velocity (top) and resolved mixture fraction (bottom) fields at the end time, $t = 0.45$ of the smooth vortex ring test case. From left to right, results are shown for the simulations $S1$, $S3$, $S5$ and the $S - LES$ simulation.

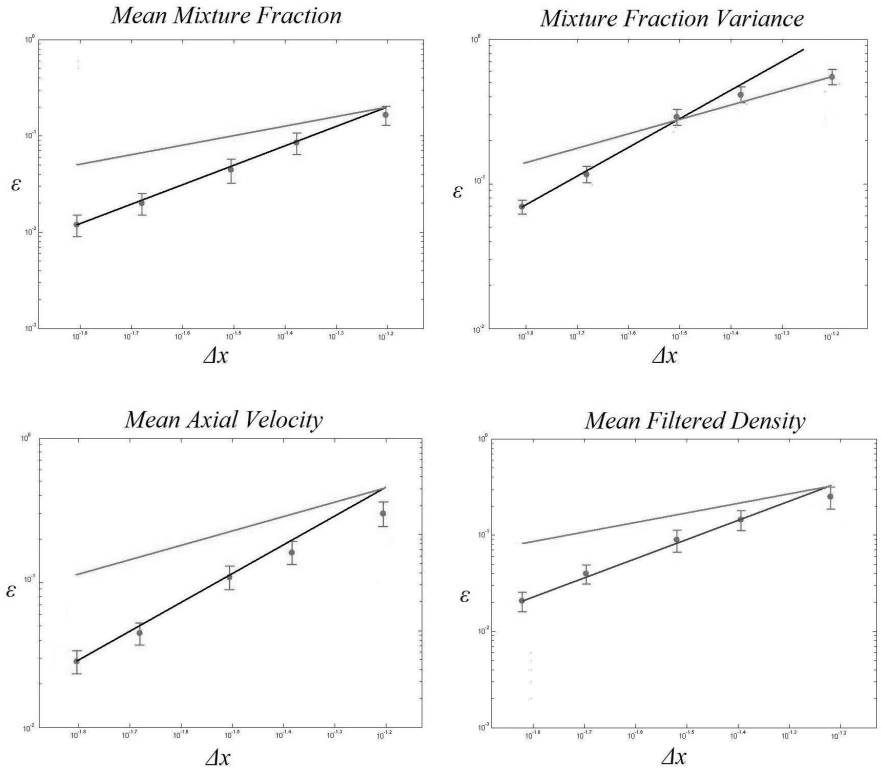

Figure 6: Mean errors and 95% confidence intervals for the convergence simulations $S1$ through $S5$. The grey and black reference lines indicate respectively first and second-order convergence behavior.

Implicit and Explicit Schemes for Mass Consistency Preservation in Hybrid Particle/Finite-Volume Algorithms for Turbulent Reactive Flows

Pavel P. Popov
Stephen B. Pope
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

Dec 10, 2012

Abstract

This work addresses the issue of particle mass consistency in Large Eddy Simulation/Probability Density Function (LES/PDF) methods for turbulent reactive flows. Numerical schemes for the implicit and explicit enforcement of particle mass consistency (PMC) are introduced, and their performance is examined in a representative LES/PDF application, namely the Sandia-Sydney Bluff-Body flame HM1. A new combination of interpolation schemes for velocity and scalar fields is found to better satisfy PMC than multilinear and fourth-order Lagrangian interpolation. A second-order accurate time-stepping scheme for stochastic differential equations (SDE) is found to improve PMC relative to Euler time-stepping, which is the first time that a second-order scheme is found to be beneficial, when compared to a first-order scheme, in an LES/PDF application. An explicit corrective velocity scheme for PMC enforcement is introduced, and its parameters optimized to enforce a specified PMC criterion with minimal corrective velocity magnitudes.

1 Introduction

In the field of turbulent reactive flow simulation, probability density function (PDF) methods have been shown to be effective in modeling turbulence-chemistry interactions [17,20,4], due to the fact that no modeling is required for the chemical source term, which is highly non-linear. Originally used as turbulence-chemistry interaction models in Reynolds-averaged Navier-Stokes

(RANS) simulations of turbulent flow, PDF methods for turbulent reactive flow are now increasingly used in conjunction with Large Eddy Simulation (LES) turbulence models [15]. The resulting LES/PDF methods, which were first introduced in [18], have been highly effective in simulating laboratory-scale reactive flows [1, 23, 27, 13].

In a typical LES/PDF simulation, the sample space of the PDF of chemical compositions has a high dimension, which makes standard, finite-difference-based methods for evolving the PDF prohibitively expensive. This necessitates the use of Monte Carlo methods for approximating the composition PDF. In this paper, we focus on the Lagrangian particle Monte Carlo approach for approximation of the composition PDF, in which an ensemble of particles, each with its own composition, is advected in physical space according to the LES Favre-averaged velocity and turbulent diffusivity, i.e., the diffusivity used to model the unresolved turbulent motions. Other Monte Carlo approximation approaches are available [16,24], each with its advantages and disadvantages over the Lagrangian particle approach.

Here, we focus on particle mass consistency (PMC), which is an important requirement in order for the Lagrangian particle ensemble to be a valid Monte Carlo approximation of the composition PDF. First identified in [12], PMC means that the expected mass of particles in a given region should equal the mass of fluid in that region as implied by the resolved density of the finite-volume (FV) solver.

Previous work on the PMC problem has included the introduction of Cartesian velocity interpolation schemes designed to give accurate values for the velocity divergence [6,10], as well as the use of a corrective velocity in the advection of particles [12,28], which reduces discrepancies in the particle mass consistency condition that accumulate due to numerical errors. Here, we extend these results by adapting the Cartesian velocity interpolation scheme of [10] to cylindrical coordinates (which are more often used for the simulation of statistically axi-symmetric canonical laboratory flames), introducing a new scheme for scalar interpolation of the turbulent diffusivity, and testing the degree of satisfaction of the PMC condition in a turbulent reactive flow representative of typical LES/PDF applications. We also employ a corrective velocity scheme, which is conceptually similar to those of [12,28], but optimized to keep mass consistency errors down to an acceptable level while minimizing the magnitude of the corrective velocities. Additionally, we examine the influence of the particle tracking SDE time integration scheme on the satisfaction of PMC.

The rest of this paper is organized as follows: in Section 2, we introduce the governing equations for an LES/PDF Lagrangian Monte Carlo simulation, and define the PMC condition. Section 3 describes the turbulent reactive flow simulation which is used to test, in conditions representative of a typical LES/PDF simulation, different schemes for the preservation of PMC. In Section 4, we compare the performance (with respect to how well PMC is satisfied) of the Euler SDE time integration scheme with that of a second-order scheme introduced by Kloeden and Platen [7]. Section 5 introduces the corrective velocity scheme for reducing PMC errors once they have accumulated due to numerical error, and

determines its optimal implementation, in terms of satisfying an appropriate PMC criterion with the least possible corrective velocity magnitude. Finally, Section 6 introduces the new interpolation schemes, and tests their performance in the PMC context. Conclusions are drawn in Section 7.

2 LES/PDF Equations and the PMC Problem

2.1 Governing Equations and the PMC Condition

We begin by defining the LES/PDF governing equations. We denote by $\overline{p}$ and $\overline{\rho}$ the LES resolved pressure and density, by $\tilde{u}_j$, $\widetilde{\nu}$ and $\tilde{D}$ the Favre-averaged resolved velocity, molecular viscosity and molecular diffusivity, and by $\tilde{S}_{ij}$ the resolved strain rate.

For the test case considered here, the molecular viscosity and diffusivity are evaluated by the power law

$$\widetilde{\nu} = \nu_0 \left(\frac{\widetilde{T}}{300K} \right)^{1.69}, \ \frac{\widetilde{\nu}}{\widetilde{D}} = \sigma, \ \sigma = 0.82, \nu_0 = 1.42 \times 10^{-5} \left[\frac{m^2}{s} \right], \tag{1}$$

where $\widetilde{T}$ denotes the resolved temperature. The resolved density is defined later in this subsection.

A turbulent viscosity and a turbulent diffusivity, $\widetilde{\nu}_T$ and $\tilde{D}_T$ respectively, are used to model the unresolved motions. The turbulent viscosity and diffusivity are evaluated by the Dynamic Smagorinsky procedure, with Δ denoting the filter size.

With these definitions, the equations of motion solved by the LES component of an LES/PDF algorithm take the form:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho} \tilde{u}_j}{\partial x_j} = 0, \tag{2}$$

$$\frac{\partial \left(\overline{\rho} \tilde{u}_j \right)}{\partial t} + \frac{\partial (\overline{\rho} \tilde{u}_i \tilde{u}_j)}{\partial x_i} = -\frac{\partial \overline{p}}{\partial x_j} + 2 \frac{\partial}{\partial x_i} \left(\overline{\rho} \left(\widetilde{\nu} + \widetilde{\nu}_T \right) \left(\tilde{S}_{ij} - \frac{1}{3} \tilde{S}_{kk} \delta_{ij} \right) \right). \tag{3}$$

The scalar mixing frequency, Ω, defined as

$$\Omega = C_\phi \frac{\tilde{D}_T + 2\tilde{D}}{\Delta^2}, \tag{4}$$

is used to evaluate turbulent mixing by the IEM mixing model, with the mixing constant set as $C_\phi = 2.0$.

In the present work material properties are a function of the composition vector, ϕ_α, which is either one-dimensional, consisting only of the mixture fraction, ξ, when a flamelet model is used, or it consists of a set of species' specific mole values, plus enthalpy. The PDF component of an LES/PDF algorithm

is a Monte Carlo approximation of the mass-weighted PDF of chemical compositions, $f(\psi;\mathbf{x},t)$, conditional on the resolved velocity field [2,21]. Here ψ_α denote points in the sample space of the composition vector ϕ_α. Using $\tilde{\phi}_\alpha(\mathbf{x},t)$ to denote the sample space mean of ϕ_α at $(\mathbf{x},t)$, the evolution equation for $f(\psi;\mathbf{x},t)$, is modeled to have the form [26]

$$\frac{\partial f}{\partial t}+\frac{\partial}{\partial x_i}\left(f\tilde{u}_i\right)=\frac{\partial}{\partial x_i}\left(\tilde{D}_T\frac{\partial f}{\partial x_i}\right)+\frac{\partial}{\partial \psi_\alpha}\left(f\Omega\left(\psi_\alpha-\tilde{\phi}_\alpha\right)\right)$$

$$-\frac{\partial}{\partial \psi_\alpha}\left(f\frac{1}{\bar{\rho}}\frac{\partial}{\partial x_i}\left(\bar{\rho}\tilde{D}\frac{\partial\tilde{\phi}_\alpha}{\partial x_i}\right)\right)+\frac{\partial}{\partial \psi_\alpha}\left(fS_\alpha(\psi)\right), \tag{5}$$

In the above equation, the left hand side accounts for transport in physical space due to the velocity and diffusivity gradients, whereas the first three terms on the right hand side of eq.(5) represent respectively turbulent diffusion using the turbulent diffusivity hypothesis, turbulent mixing using the IEM mixing model, and molecular diffusion, and $S_\alpha(\boldsymbol{\psi})$ is the reaction source term ($S_\alpha(\boldsymbol{\psi})=0$ when a flamelet model is used). The advantage of modeling chemistry by using a mass-weighted composition PDF which evolves by eq.(5) is that the source term, which is highly non-linear as a function of $\boldsymbol{\psi}$, requires no modeling. Additionally, in the DNS limit, when $\tilde{D}_T=0$, the results yielded by the present specification of the PDF are consistent with an exact solution of the Navier-Stokes and chemical transport equations [9].

With this definition of the mass-weighted PDF, the resolved specific volume, $\tilde{v}$, is defined by

$$\tilde{v}(\mathbf{x},t)=\int v(\boldsymbol{\psi})f(\boldsymbol{\psi};\mathbf{x},t)\,|\mathrm{d}\boldsymbol{\psi}|\,, \tag{6}$$

where $v(\boldsymbol{\psi})$ is the constitutive relation for specific volume as a function of the composition variables. Finally, the resolved density is defined as

$$\bar{\rho}=1/\tilde{v}. \tag{7}$$

An additional scalar which we solve for is the transported specific volume, $\hat{v}$, defined to evolve by the equation

$$\frac{\partial\left(\overline{\rho}\hat{v}\right)}{\partial t}+\frac{\partial\left(\overline{\rho}\widetilde{u}_j\hat{v}\right)}{\partial x_j}=\frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{D}_T\frac{\partial\hat{v}}{\partial x_j}\right)+S_v+\omega_v, \tag{8}$$

where S_v and ω_v are source and relaxation terms [22]. In particular, introducing a relaxation time τ (which is in this study set to $4\Delta t$) the relaxation term is defined as

$$\omega_v=\bar{\rho}\frac{\tilde{v}-\hat{v}}{\tau}, \tag{9}$$

which insures that $\hat{v}$ tends to $\tilde{v}$ in the limit when τ goes to zero and the number of particles per cell goes to infinity.

In order to efficiently calculate the evolution of eq.(5), without having to perform a discretization on the high-dimensional composition space, we perform a Monte Carlo approximation [17]. An ensemble of Lagrangian particles are initialized throughout the computational domain and evolve by a set of stochastic differential equations (SDEs). Throughout this work, we shall use the superscript * to denote particle quantities, as opposed to Eulerian fields (such as $\tilde{u}_j$ and $\overline{\rho}$); we shall also use it to denote Eulerian fields interpolated at a given particle's current location.

Thus, each particle has a mass m^*, determined at its initialization to correspond to the mass of fluid in the region taken up by the particle, a location X_j^* which evolves according to the LES velocity and turbulent diffusivity, and a composition vector, ϕ_α^*. The evolution equations for the particle position and composition vectors are as follows:

$$dX_j^* = \left[\widetilde{u}_j + \frac{1}{\overline{\rho}}\frac{\partial\left(\overline{\rho}\widetilde{D}_T\right)}{\partial x_j}\right]^* dt + \left[2\tilde{D}_T^*\right]^{1/2} dW_j^*, \tag{10}$$

$$d\phi_\alpha^* = -\Omega^*\left(\phi_\alpha^* - \widetilde{\phi}_\alpha^*\right)dt + \left[\frac{1}{\overline{\rho}}\frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{D}\frac{\partial\widetilde{\phi}_\alpha}{\partial x_j}\right)\right]^* dt + S_\alpha\left(\boldsymbol{\phi}^*\right)dt. \tag{11}$$

where the term dW_j^* in eq.(10) denotes a Wiener increment, and the three terms on the right hand side of eq.(11) denote respectively turbulent mixing (in this case represented by the IEM model [25]), molecular diffusion, and chemical reaction.

Using the notation $\langle\cdot\rangle$ to denote expectation over all possible initial particle locations and all possible realizations of the Wiener process, we define the particle mass-weighted PDF, $f^*\left(\boldsymbol{\psi};\mathbf{x},t\right)$ by:

$$f^*\left(\boldsymbol{\psi};\mathbf{x},t\right) = \frac{\left\langle m^*\delta\left(\mathbf{X}^*(t)-\mathbf{x}\right)\delta\left(\boldsymbol{\phi}^*(t)-\boldsymbol{\psi}\right)\right\rangle}{\left\langle m^*\delta\left(\mathbf{X}^*(t)-\mathbf{x}\right)\right\rangle}, \tag{12}$$

and, in the absence of numerical errors, eqs.(10,11) imply that f^* evolves by eq.(5), so that, provided that $f = f^*$ initially and at the boundaries, the particle mass-weighted PDF f^* is equal to the mass-weighted PDF:

$$f\left(\boldsymbol{\psi};\mathbf{x},t\right) = f^*\left(\boldsymbol{\psi};\mathbf{x},t\right), \tag{13}$$

due to the fact that both PDF functions have the same evolution equation, initial and boundary conditions.

An additional quantity which we introduce at this point is the particle mass density, q, which is defined as the expected mass density of particles at $\mathbf{x}$,

$$q(\mathbf{x},t) \equiv \left\langle m^*\delta\left(\mathbf{X}^*(t)-\mathbf{x}\right)\right\rangle. \tag{14}$$

The main focus of this paper is in the preservation of the consistency,

$$q = \overline{\rho}, \tag{15}$$

between particle mass density and LES resolved density: we shall refer to eq.(15) as the PMC condition. As pointed out by [12,28], eq.(15) is one of the essential conditions for the correspondence between the PDFs f and f^* which we use in our Monte Carlo approximation. In the absence of numerical errors the consistency condition of eq.(15) is an identity because eq.(2) can be simply rearranged to yield

$$\frac{\tilde{D}\ln(\bar{\rho})}{\tilde{D}t} = -\frac{\partial \tilde{u}_j}{\partial x_j}, \tag{16}$$

where $\frac{\tilde{D}}{\tilde{D}t}$ denotes a convective derivative with velocity $\tilde{u}_j$, whereas eq.(10) implies that q evolves by

$$\frac{D^*\ln(q)}{Dt^*} = -\frac{\partial \tilde{u}_j^*}{\partial x_j}, \tag{17}$$

where $\frac{D^*}{Dt^*}$ denotes a convective derivative with velocity $\tilde{u}_j^*$. Therefore, in the absence of numerical errors, the evolution equations, eqs.(16,17) imply $q = \bar{\rho}$ for all time provided that $q = \bar{\rho}$ at $t = 0$ and at inflow boundaries.

2.2 Numerical Implementation

The LES equations are solved on a finite volume (FV) grid, with x, r, θ denoting respectively the axial, radial and azimuthal coordinates [14]. The domain is also divided into a set of PDF cells, each of which contains approximately n_{pc} particles, where n_{pc} is a specified parameter. The PDF cells are identical to the LES cells away from the centerline; close to the centerline, one PDF cell consists of several LES cells, concatenated in the azimuthal direction [27].

The ensemble of particles evolving by eqs.(10,11) is used to determine local cell values as mass-weighted sums. More specifically, let $B(\mathbf{x})$ be a given cell basis function: in the present work, $B(\mathbf{x})$ are continuous tent functions, piecewise linear in each of the x, r, θ coordinates, which form a partition of unity. For a particular cell, we have:

$$B(\mathbf{x}) = B_x(x)\, B_r(r)\, B_\theta(\theta), \tag{18}$$

where, denoting by x_0 the axial location of the cell's midpoint, and denoting by x_1, x_{-1} the axial locations of its axial neighbors, $B_x(x)$ is defined as

$$B_x(x) = \max\left(0, \min\left(\frac{x - x_1}{x_0 - x_1}, \frac{x - x_{-1}}{x_0 - x_{-1}}\right)\right), \tag{19}$$

and $B_r(r), B_\theta(\theta)$ are similarly defined.

For a specific particle property y, the Favre mean of y at the location of a given LES/PDF cell is approximated as

$$\tilde{y} \approx \frac{\sum_{cell} m^* y^* B\left(\mathbf{X}^*(t)\right)}{\sum_{cell} m^* B\left(\mathbf{X}^*(t)\right)}, \tag{20}$$

where the summation on the right-hand side is over all particles which are currently in the support of $B(\mathbf{x})$, which consists of the cell on which $B(\mathbf{x})$ is centered, and parts of its immediate neighbors in each of the x, r, θ directions

This methodology for cell value estimation is used to communicate information from the PDF to the LES component of the LES/PDF algorithm. For example, a straightforward method for evaluating the resolved density, $\overline{\rho}$, used in eqs.(2,3), is to just take the inverse of the Favre-averaged specific volume, $\tilde{v}$. In the present work, $\overline{\rho}$ is estimated by another method, called the transported specific volume approach, which reduces the amount of statistical error (see [22] for more details). In that approach, we use the additional transported scalar $\hat{v}$, to be solved for by the LES code, and use the inverse of that specific volume as an approximation of the resolved density.

$$\hat{\rho} \equiv \frac{1}{\hat{v}} \approx \bar{\rho}. \tag{21}$$

As previously pointed out, eqs.(8,9) imply that, for the present choice of $\tau = 4\Delta t$, $\hat{v}$ tends to $\tilde{v}$ in the limit as Δt goes to zero and the number of particles per cell goes to infinity, and hence using eq.(21) to obtain the resolved density yields a valid numerical approximation to the system of equations defined in the previous subsection.

In the present work, q is approximated on LES/PDF cells by the formula

$$q \approx \frac{\sum_{cell} m^* B\left(\mathbf{X}^*\right)}{\int_{cell} B(\mathbf{x}) \left|\mathrm{d}\mathbf{x}\right|}, \tag{22}$$

where $\int_{cell}$ is an integral over the support of $B(\mathbf{x})$.

As previously noted, in the absence of numerical error $q = \overline{\rho}$ exactly. However, in a practical LES/PDF simulation, differences between q and $\overline{\rho}$ will develop, and need to be kept at a low level so that the combination of the LES (eqs.(2,3)) and PDF (eqs.(10,11)) components of the code yields a valid approximation to the mass-weighted composition PDF.

Here, we introduce a PMC error variable ϵ^c, defined by

$$\epsilon^c = q/\bar{\rho} - 1, \tag{23}$$

to quantify the difference between q and $\bar{\rho}$, and we rank the performance of different PMC preservation schemes by the approximate L^1 and L^∞ measures of ϵ^c, defined by

$$\left\|\epsilon^c\right\|_1 = \frac{\sum_{i,m} \left|\epsilon_i^c(t_m)\right| V_i \Delta t}{\sum_{i,m} V_i \Delta t} \tag{24}$$

and

$$\|\epsilon^c\|_\infty = \max_{i,m} |\epsilon_i^c(t_m)| \tag{25}$$

where i the index of a given LES cell, t_m denotes the midpoint of a given time step in the second half of the simulation, and $\epsilon_i^c(t_m)$ and V_i denote respectively the value of ϵ^c for a given cell at time t_m, and the volume of that cell.

3 Description of the Bluff-Body Jet Simulation

Here, we describe the turbulent reactive flow simulation used as a test case for different PMC preservation schemes.

We simulate the Sandia-Sydney Bluff-Body Flame HM1, a detailed description of which can be found in Masri and Bilger [8]. The flame consists of a jet of diameter $3.6mm$ inside a bluff body of diameter $50mm$, which is itself located inside a square wind tunnel whose sides are $150mm$. The jet is a $1:1$ by volume mixture of CH_4 and H_2 at a temperature of $300K$, and leaves the outlet at a bulk velocity of $118m/s$ – the Reynolds number, based on the jet velocity and diameter, is $29,900$. The coflow consists of air, also at $300K$, with a bulk velocity of $40m/s$. The computational domain is $x \in [0, 20R_B]$, $r \in [0, 3.39R_B]$, where R_B is the bluff-body radius. There is a wall boundary at $r = 3.39R_B$, which yields the same hydraulic diameter as the wind tunnel used in the experiment. The grid size is $96 \times 64 \times 48$, which is not sufficient for an accurate simulation of the bluff-body, but allows us to test the PMC preservation properties of a number of different numerical schemes in a flow representative of LES/PDF applications.

In the present simulation, we use the GRI 1.2 chemical mechanism with chemical composition being advanced either by steady flamelet modeling, for which the composition vector ϕ_α^* consists only of the mixture fraction ξ, or via integration in the full composition space, in which case ϕ_α^* is $33-$dimensional - we shall use the abbreviation FC from now on to refer to the full composition space simulations. In FC simulations, *in situ* adaptive tabulation (ISAT) [19] is used for speed up of the integration of the chemical source term. The number of particles per cell is either $n_{pc} = 30$ or $n_{pc} = 50$, and the time step is $\Delta t = 8.48\mu s$. Simulations are run for 12 flow-through times based on the coflow velocity (which amounts to 20000 time steps), by the latter half of which period the flow has reached a statistically stationary state: the results presented are based on statistics collected during the second half of the simulation interval.

Figures 1-4 show contour plots of the Favre means of the velocity and temperature fields, as well as plots of the instantaneous fields at the end of the simulation. As can be seen, the flow is highly non-trivial, with a recirculation region extending up to a distance $2R_B$ downstream from the bluff body, and considerable fluctuations about the mean. This makes it a good test case to study the problem PMC, which depends considerably on the evolution of particles in physical space.

In the subsequent sections, we present results for this test case with different implicit and explicit schemes of preservation of PMC.

4 SDE Time Integration as an Implicit Scheme for PMC Preservation

First, we examine the effect on PMC of the numerical scheme used to integrate the position advection SDE, eq.(10). The standard Euler SDE time integration scheme is most commonly used [3,5] in particle/FV algorithms for turbulent reactive flows; those studies which use a SDE integration scheme with a higher order of accuracy [27] have detected little improvement in the overall solution, for test cases which are representative of typical LES/PDF turbulent flows. Here, we compare the performance of the Euler scheme with that of the weakly second-order accurate derivative-free SDE integration scheme introduced by Kloeden and Platen [7], from here on referred to as the KP scheme, for the sake of brevity.

Table 1 gives a PMC error comparison between simulations using Euler and the KP time integration schemes. The simulation is as described in section 3 - both the flamelet and FC calculations are used. The value of n_{pc} used is $n_{pc} = 30$, and the interpolation scheme for the velocity and scalar fields is standard multilinear interpolation; no explicit PMC correction algorithms are used. Also given is the computational cost, in overall simulation wall clock time per particle per step - the total number of particles is approximated as n_{pc} times the number of grid cells.

As can be seen on the table, in both the flamelet and FC simulations, the use of the KP scheme instead of Euler reduces both the L^1 and L^∞ errors by approximately 30%, for a modest increase in computational cost, especially for the FC simulations. Thus we see that, in contrast to previous experience (in which PMC errors were not examined) with particle/FV methods, it is advantageous to use a higher-order SDE integration scheme. It can also be seen from Table 1 that, with respect to PMC errors, there is little difference between the flamelet and FC simulations.

Figure 5 gives a detailed look at the PMC error variable, ϵ^c, throughout the domain, for the FC simulation with Euler time-stepping - the results for the other simulations are similar, the main difference being in the magnitude of ϵ^c. As can be seen on fig. 5, while there is considerable noise in the instantaneous values of ϵ^c, there is also a deterministic component in the ϵ^c field, which can be seen on the time-averaged contour plot. At both the jet and coflow shear layers, we see a region with positive $\langle \epsilon^c \rangle_T$, i.e. a greater mass density of particles than there should be, on the bluff-body side of the shear layers, and a region with negative $\langle \epsilon^c \rangle_T$ at the inlet sides of the shear layers.

Table 1 also provides L^1 and L^∞ measures of these time-averaged values, denoted by $\langle \epsilon^c \rangle_T$, for the four simulations.

Chemistry/ SDE scheme	Flamelet/Euler	Flamelet/KP	FC/Euler	FC/KP
$\|\epsilon^c\|_1$	1.54×10^{-1}	1.09×10^{-1}	1.67×10^{-1}	1.16×10^{-1}
$\|\epsilon^c\|_\infty$	5.78×10^{-1}	3.97×10^{-1}	5.82×10^{-1}	3.95×10^{-1}
$\| \langle\epsilon^c\rangle_T \|_1$	1.88×10^{-2}	1.05×10^{-2}	2.32×10^{-2}	1.03×10^{-2}
$\| \langle\epsilon^c\rangle_T \|_\infty$	2.12×10^{-1}	1.22×10^{-1}	2.14×10^{-1}	1.20×10^{-1}
cost/ptcl	$10.2\mu s$	$13.3\mu s$	$81.7\mu s$	$84.2\mu s$

Table 1: L^1 and L^∞ measures, $\|\epsilon^c\|_1$ and $\|\epsilon^c\|_\infty$, of the instantaneous PMC error, L^1 and L^∞ measures, $\| \langle\epsilon^c\rangle_T \|_1$ and $\| \langle\epsilon^c\rangle_T \|_\infty$, of the time-averaged error $\langle\epsilon^c\rangle_T$, and computational cost in wall clock time per particle per step per core, for simulations with multilinear interpolation and no explicit mass correction.

As we can see, the KP scheme yields approximately a 45% reduction in the time-averaged, deterministic components of PMC error. Also, we note again that, with respect to PMC errors, there is little difference between flamelet and FC simulations.

The mechanism for accumulation of deterministic PMC errors is replicated in a simple 2D numerical test case, described in Appendix A. In that test case, the only source of error is due to the time-stepping scheme, which allows us to isolate the PMC errors due to time stepping, and evaluate the performance of different SDE integration schemes, apart from the other components of an LES/PDF solution. The 2D test flow described in the Appendix is a model for the bluff-body shear layer: it contains an interface between regions of high and low axial velocity; this interface coincides with a region of increased turbulent diffusivity, similarly to the flow features seen on fig. 6. The results are shown on figures 7 and 8; as can be seen on these figures, there is a buildup of deterministic mass-consistency errors, with positive $\langle\epsilon^c\rangle_T$ in the region of low axial velocity, and negative $\langle\epsilon^c\rangle_T$ in the region of high axial velocity. This distribution is similar to the distribution of deterministic errors seen on fig. 5, which suggests that the latter are in large part caused by time-stepping error. This explains the reduction of deterministic PMC error seen on table 1, as it can be seen on figures 7 and 8 that the KP scheme greatly reduces the deterministic PMC errors due to time stepping.

From the above results we conclude that, despite the modest increase in computational cost, from a PMC standpoint it is advantageous to use an SDE time integration scheme with a higher order of accuracy than the Euler scheme. In the following sections, unless otherwise specified, it is implicit that the KP scheme is used.

5 Explicit PMC Preservation Schemes: Velocity Correction

In this section, we describe and test a scheme for the reduction of PMC errors once they have accumulated due to numerical error.

5.1 Determining the Desired Maximum Level of PMC Error

Before we describe the explicit PMC correction schemes used here, let us consider the desired level, ϵ_0^c, of PMC error which we wish to enforce. Due to the stochastic nature of LES/PDF codes, it is undesirable to completely reduce ϵ^c to zero - for a given value of n_{pc} (which is typically in the range of 20 to 50 for most practical simulations) the error ϵ^c will contain a component proportional to $n_{pc}^{-1/2}$, which is due to the finite sample size in the approximation of q and does not in itself imply a discrepancy between q and $\overline{\rho}$. This is true even with perfect integration of the SDE in eq.(10), which occurs for example when we have constant velocity and no turbulent diffusivity, so that the interpolation and SDE integration schemes contain no numerical errors; in this idealized case, $q = \overline{\rho}$ is satisfied exactly, and ϵ^c contains only sampling error which should be ignored.

Therefore, we shall set ϵ_0^c to be equal to the maximal rms ϵ^c error for a case with perfect integration of eq.(10). For simplicity, let us consider a uniform Cartesian grid whose grid cells have volume 1, assume that the velocity is constant throughout the domain, that there is no diffusivity and that the basis functions $B(\mathbf{x})$ used for evaluation of cell means are the indicator functions of a given cell. Also, let us assume that the density $\overline{\rho}$ is constant, and that all particles have the same mass, so that $q/\overline{\rho}$ for a given cell can be evaluated as the number of particles in that cell, divided by n_{pc}.
At the beginning of a simulation, exactly n_{pc} particles are initialized with their position having uniform probability in each cell, so that $q/\overline{\rho} = 1$, hence $\epsilon_0^c = 0$. However, as the simulation progresses and the particle distribution shifts with respect to the FV grid, each FV grid cell will cover a region of particles which were initialized in 8 separate grid cells (i.e. 2^3 grid cells for the 3D case considered). Let us consider a given cell, C, and denote by V_1 through V_8 the volumes of the intersections between that cell and the Lagrangian mappings for the present time of the 8 cells, C_1 through C_8, whose particles currently reside in C. Due to the uniform initialization of particles and the fact that we have chosen all cells to have volume 1, we have that the probability that a particle initialized in C_i currently lies in C is equal to V_i, with $\sum_{i=1}^{8} V_i = 1$. Therefore, if we denote by N the number of particles currently in C, we get that N the sum of 8 binomial random variables with parameters n_{pc} and V_i, respectively. Hence, since $\sum_{i=1}^{8} V_i = 1$, the expectation of N is $E(N) = n_{pc}$, and the variance of N

is

$$\mathrm{Var}\,(N) = n_{pc} \sum_{i=1}^{8} V_i\,(1 - V_i). \tag{26}$$

Given the constraint $\sum_{i=1}^{8} V_i = 1$, for a fixed n_{pc} the right hand side of eq.(26) attains its maximum for $V_1 = V_2 = \cdots = V_8 = 1/8$ (from the inequality between arithmetic and quadratic means), which gives us that the maximal value for $\mathrm{Var}\,(N)$ is $\mathrm{Var}\,(N) = 7/8n_{pc}$, and so the maximal possible standard deviation for $q/\overline{\rho} = N/n_{pc}$ for this simplified case is

$$\max\left(\mathrm{std}\left(\frac{q}{\overline{\rho}}\right)\right) = \left(\frac{7}{8n_{pc}}\right)^{1/2}. \tag{27}$$

Based on this, we set

$$\epsilon_0^c = \left(\frac{7}{8n_{pc}}\right)^{1/2} \tag{28}$$

as the maximal allowable consistency error, i.e. we require that $\|\epsilon^c\|_\infty < \epsilon_0^c$. For a criterion on $\|\epsilon^c\|_1$, we define ϵ_1^c to be the expectation of $|N/n_{pc} - 1|$:

$$\epsilon_1^c = E\,(N/n_{pc} - 1)\,, \tag{29}$$

which is the L^1 counterpart of the variance-based (i.e., using an L^2 norm) criterion of eq.(28), and we require that $\|\epsilon^c\|_1 < \epsilon_1^c$. For $n_{pc} = 30, 50$, ϵ_1^c has the values $0.136, 0.105$ respectively (obtained numerically). As we shall see below $\|\epsilon^c\|_\infty < \epsilon_0^c$ is a more restrictive criterion than $\|\epsilon^c\|_1 < \epsilon_1^c$

5.2 Corrective Schemes for the Reduction of ϵ^c

As seen in the previous section, even though using the KP time integration scheme reduces the magnitude of the PMC errors, those errors are still considerably higher than the desired level ϵ_0^c introduced in eq.(28) of the previous subsection - for $n_{pc} = 30$ we have $\epsilon_0^c = 0.1708$, whereas the smallest value of $\|\epsilon^c\|_\infty$ obtained in the previous section is $\|\epsilon^c\|_\infty = 0.397$. This necessitates the use of a correction algorithm to reduce PMC errors down to a desired level. Such correction algorithms typically take the form of velocity correction algorithms [28,12]. In this work, we also use a velocity correction algorithm similar to that of [28].

In particular, we introduce a corrective velocity, u_j^c, as a discrete FV field, and we set

$$\tilde{u}_j^* = \tilde{u}_j + u_j^c. \tag{30}$$

Then, making the assumption that $\|u_j^c\| \ll \|\tilde{u}_j\|$ and that interpolation errors are small, we get that $\frac{\tilde{D}}{\tilde{D}t} \approx \frac{D^*}{Dt^*}$, and so eqs.(16,17) yield that

$$\frac{\tilde{D}\ln(q/\bar{\rho})}{\tilde{D}t} \approx \frac{D^*\ln(q)}{Dt^*} - \frac{\tilde{D}\ln(\bar{\rho})}{\tilde{D}t} = -\frac{\partial u_j^c}{\partial x_j}. \tag{31}$$

Then, assuming that $\epsilon^c \ll 1$, so that $\frac{q}{\bar{\rho}} \approx 1$, we get that

$$\frac{\tilde{D} \ln (q/\bar{\rho})}{\tilde{D}t} = \frac{\bar{\rho}}{q} \frac{\tilde{D} (q/\bar{\rho})}{\tilde{D}t} \approx \frac{\tilde{D}\epsilon^c}{\tilde{D}t}, \tag{32}$$

and hence

$$\frac{\tilde{D}\epsilon^c}{\tilde{D}t} \approx -\frac{\partial u_j^c}{\partial x_j}, \tag{33}$$

which means that setting $\frac{\partial u_j^c}{\partial x_j}$ to be proportional to ϵ^c will enforce a decay of ϵ^c toward zero. More specifically, we introduce a velocity potential, ζ, so that

$$u_j^c = \frac{\partial \zeta}{\partial x_j}, \tag{34}$$

and we require ζ to satisfy the Poisson equation

$$\frac{\partial u_j^c}{\partial x_j} = \frac{\partial^2 \zeta}{\partial x_i \partial x_i} = \frac{F(\epsilon^c)}{\tau^c}, \tag{35}$$

where on the right-hand side of the above equation: $F(\cdot)$ is an operator acting on the ϵ^c field, the simplest example being $F(\epsilon^c) = \epsilon^c$; and where τ^c is a control parameter with units of time, which manifests as the time scale of the decay of ϵ^c toward zero.

A similar definition of the corrective velocity u_j^c has been introduced previously, by Zhang and Haworth [28], in which the authors use $F(\epsilon^c) = \epsilon^c$, and vary the value of τ^c. The main contribution of this work in the area of explicit mass correction algorithms is in testing alternative definitions of $F(\epsilon^c)$, and establishing which are optimal for the purpose of maintaining the consistency error below the desired level ϵ_0^c, while keeping the magnitude of the corrective velocity to a minimum.

We use three different definitions of $F(\epsilon^c)$. The first, denoted $F_1(\epsilon^c)$ is the straightforward choice

$$F_1(\epsilon^c) = \epsilon^c. \tag{36}$$

In the second choice, spatial smoothing is applied to the the field ϵ^c: the algorithm used is the implicit 3-point smoothing algorithm, described in Viswanathan et al. [26], in which the amount of spatial smoothing performed on fields is controlled by a parameter α, which is defined so that the amount of the resultant implicit smoothing is equivalent to explicit smoothing over α cells in each direction (in particular $\alpha = 1$ implies that no smoothing is performed). Using the operator $S^\alpha[\cdot]$ to denote the smoothing procedure, we define $F_2^\alpha(\epsilon^c)$ as

$$F_2^\alpha(\epsilon^c) = S^\alpha[\epsilon^c]. \tag{37}$$

This choice of $F_2^\alpha(\epsilon^c)$ is motivated by the reasoning that, by ignoring the highest wavenumber components of ϵ^c, the resulting correction algorithm will yield, for

the same values of the control parameter τ^c, lower magnitudes of the corrective velocity u_j^c, thus allowing us to use smaller values of the decay parameter τ^c.

Finally, we also consider a definition, $F_3(\epsilon^c)$ which aims to reduce the amount of corrective velocity by only correcting PMC errors when they become significant compared to ϵ_0^c. In particular, $F_3(\epsilon^c)$ is defined by

$$F_3(\epsilon^c) = \begin{cases} 0, \text{ for } |\epsilon^c| < \epsilon_0^c/2 \\ \epsilon^c \sin^2\left(\frac{\pi(|\epsilon^c| - \epsilon_0^c)}{\epsilon_0^c}\right), \text{ for } \epsilon_0^c/2 \le |\epsilon^c| < \epsilon_0^c \\ \epsilon^c, \text{ for } |\epsilon^c| \ge \epsilon_0^c \end{cases}, \tag{38}$$

that is, between $|\epsilon^c| = \epsilon_0^c/2$ and $|\epsilon^c| = \epsilon_0^c$, F_3 varies smoothly between $F_3 = 0$ (for $|\epsilon^c| \le \epsilon_0^c/2$) and $F_3 = \epsilon_c$ (for $|\epsilon^c| \ge \epsilon_0^c$).

The overall performance of the three corrective strategies is shown on Table 2 below. The first three rows show results for a simulation with $n_{pc} = 30$, the last three for a simulation with $n_{pc} = 50$. The table compares L^1 and L^∞ measures of the corrective velocity, respectively denoted as $\left\|\epsilon^U\right\|_1$ and $\left\|\epsilon^U\right\|_{L\infty}$ (where ϵ^U is a dimensionless velocity error normalized by the maximal flow velocity), with the PMC errors produced by each scheme. The control parameter τ^c is set to have the largest possible value while still maintaining $\|\epsilon^c\|_\infty < \epsilon_0^c$ and $\|\epsilon^c\|_1 < \epsilon_1^c$, where $\epsilon_0^c = 0.171, \epsilon_1^c = 0.136$ and $\epsilon_0^c = 0.132, \epsilon_1^c = 0.105$, respectively for $n_{pc} = 30$ and $n_{pc} = 50$. Due to the large number of simulations required to explore the parameter space for τ^c and α, all of the results presented in this section are obtained from simulations with flamelet chemistry modeling.

Corrective Scheme	$\tau^c/\Delta t$	n_{pc}	$\|\epsilon^c\|_1$	$\|\epsilon^c\|_\infty$	$\|\epsilon^U\|_1$	$\|\epsilon^U\|_\infty$
F_1	8	30	4.4×10^{-2}	1.53×10^{-1}	9.1×10^{-2}	1.58×10^{-1}
F_3	8	30	5.1×10^{-2}	1.60×10^{-1}	8.1×10^{-2}	1.49×10^{-1}
$F_2^{\alpha=3.0}$	4	30	4.6×10^{-2}	1.52×10^{-1}	4.5×10^{-2}	9.7×10^{-2}
F_1	10	50	3.5×10^{-2}	1.29×10^{-1}	7.4×10^{-2}	1.22×10^{-1}
F_3	10	50	4.4×10^{-2}	1.31×10^{-1}	6.7×10^{-2}	1.19×10^{-1}
$F_2^{\alpha=3.0}$	6	50	3.7×10^{-2}	1.31×10^{-1}	3.8×10^{-2}	7.9×10^{-2}

Table 2: Comparison between the three corrective schemes, for flamelet simulations with $n_{pc} = 30$ and $n_{pc} = 50$

As we can see on the table, the criteria $\|\epsilon^c\|_1 < \epsilon_1^c, \|\epsilon^c\|_\infty < \epsilon_0^c$ can be enforced by all three schemes; also note that, as mentioned above $\|\epsilon^c\|_\infty < \epsilon_0^c$ is the more restrictive criterion. Contrary to expectations, the F_3 corrective scheme does not yield considerable improvement over F_1: the L^1 measure of the PMC error increases by approximately 20%, while the L^1 measure of the corrective velocity decreases by approximately 10%, and the difference in the L^∞ error measures is even less.

In contrast, the F_2 scheme with $\alpha = 3.0$, and a smaller value for Δt gives considerable improvements over F_1 - the consistency errors are similar, whereas the corrective velocity errors are reduced by approximately 50% in the L^1 sense and 35% in the L^∞ sense. The choice of $\alpha = 3.0$ was arrived at by varying α from 1.0 to 4.0, and τ^c from its F_1 value down to $\tau^c = 2\Delta t$. The L^1 results of this parameter study are shown on Table 3, for $n_{pc} = 30$.

τ^c	$\|\epsilon^c\|_1 / \|\epsilon^U\|_1$, $\alpha = 1.0$	$\|\epsilon^c\|_1 \|\epsilon^U\|_1$, $\alpha = 2.0$	$\|\epsilon^c\|_1 \|\epsilon^U\|_1$, $\alpha = 3.0$	$\|\epsilon^c\|_1 \|\epsilon^U\|_1$, $\alpha = 4.0$
2.0	0.027 | 0.246	0.034 | 0.120	0.038 | 0.070	0.041 | 0.060
4.0	0.033 | 0.139	0.040 | 0.065	0.046 | 0.047	0.049 | 0.039
6.0	0.039 | 0.102	0.049 | 0.056	0.051 | 0.036	0.056 | 0.035
8.0	0.044 | 0.091	0.057 | 0.048	0.059 | 0.032	0.065 | 0.027

Table 3: L^1 consistency and corrective velocity errors for $n_{pc} = 30$ - a parameter study for α and τ^c.

As we can see $\alpha = 3.0$, $\tau^c = 4\Delta t$ yields the least corrective velocity for the same (or less) consistency error as in the $F_1(\cdot)$ case with $\tau^c = 8\Delta t$, which is the greatest value of τ^c such that the criterion of eq. (28) is satisfied. The results are similar for $n_{pc} = 50$ - the optimal combination is $\alpha = 3.0$, $\tau^c = 6\Delta t$. Henceforth, we shall use F_2 with these parameters as our explicit PMC correction algorithm.

Note that in a different LES/PDF simulation the value of the control parameter τ^c required to maintain $\|\epsilon^c\|_1 < \epsilon_1^c, \|\epsilon^c\|_\infty < \epsilon_0^c$ may be different; as the test case in Appendix A shows for example, the PMC errors are a function of the turbulent diffusivity magnitude. Therefore, when applying the $F_2^{\alpha=3.0}$ corrective scheme to a new flow, the user is encouraged to monitor the levels of $\|\epsilon^c\|_1, \|\epsilon^c\|_\infty$ in the initial stages of the simulation, and adjust τ^c accordingly - reduce it if the PMC errors are above the desired level, increase it if they are below the desired level.

6 Velocity and Scalar Interpolation Schemes as Implicit Methods for Preserving PMC

In this section, we introduce new interpolation schemes for the velocity and scalar fields in eq.(10), which are designed to reduce PMC errors, both in terms of reducing ϵ^c for simulations without explicit PMC correction, and in terms of reducing the corrective velocity imposed by the correction algorithms discussed in the previous section. We then compare the performance of these new schemes with that of standard multilinear and fourth-order Lagrangian interpolation.

6.1 Velocity Interpolation: the Polar Parabolic Edge Reconstruction Method

The Polar Parabolic Edge Reconstruction Method (PPERM) is an adaptation for cylindrical grids of a previous velocity interpolation scheme, called the Parabolic Edge Reconstruction Method (PERM), introduced by McDermott and Pope [10]. As noted by the authors of that paper, the combination of the evolution equations for $\bar{\rho}$ and q, respectively eq.(16,17), implies that, with $\bar{\rho} = q$ at the beginning of the simulation (which is always the case, provided the particles are initialized correctly), the magnitude of the differences between $\bar{\rho}$ and q is proportional to the magnitude of the differences between $\tilde{u}_j$ and $\tilde{u}_j^*$, and between $\frac{\partial \tilde{u}_j}{\partial x_j}$ and $\frac{\partial \tilde{u}_j^*}{\partial x_j}$. Therefore, a velocity interpolation scheme must yield accurate values for the velocity $\tilde{u}_j^*$ and its divergence, $\frac{\partial \tilde{u}_j^*}{\partial x_j}$, in order to maintain PMC.

To this end, PERM, which is an extension of multilinear interpolation with additional polynomial terms, is a second-order accurate (with respect to the grid spacing) velocity interpolation scheme, with second-order accurate values for the divergence for the interpolated velocity. This is an advantage over multilinear interpolation, whose divergence of the velocity is only first-order accurate. An additional advantage of PERM over multilinear interpolation is that for an FV discrete velocity field $\tilde{u}_j$ which is discretely divergence-free (i.e., the total velocity flux through the faces of each FV cell is 0), the PERM interpolated velocity is divergence-free at all points in the domain.

The new interpolation scheme, PPERM, is an adaptation of PERM to cylindrical coordinate grids which retains these properties. A detailed description of PPERM, the process for its evaluation, and numerical tests of its convergence properties are given in Appendix B.

6.2 Diffusivity and Density Interpolation: the Multilinear Gradients Method

The new Multilinear Gradients Method (MLG) introduced here is a third-order accurate interpolation scheme for scalar fields - in this work, it is applied to $\bar{\rho}\tilde{D}_T$ and $\bar{\rho}$ in eq.(10). The MLG-interpolated fields are piecewise polynomial in x, r, and θ. The 20 polynomial terms used are of up to fourth degree, and comprise the minimal set necessary so that the gradient of the MLG reconstructed field has the same functional form as when the gradient is itself interpolated by a multilinear velocity interpolation scheme. This is done so that the value of $\frac{\partial\left(\bar{\rho}\tilde{D}_T\right)}{\partial x_j}$ used in eq.(10) is second-order accurete with respect to the grid spacing.

Note that second-order accuracy of $\frac{\partial\left(\bar{\rho}\tilde{D}_T\right)}{\partial x_j}$ can also be achieved by performing multilinear interpolation of $\frac{\partial\left(\bar{\rho}\tilde{D}_T\right)}{\partial x_j}$ as a vector field, separately from the interpolation of the scalars $\bar{\rho}, \bar{\rho}\tilde{D}_T$. The advantages of using MLG is that the additional computational work performed in obtaining second-order accurate

values for the gradient is also utilized in obtaining third-order accurate scalar fields, and that the interpolated value for $\frac{\partial\left(\bar{\rho}\tilde{D}_T\right)}{\partial x_j}$ is in fact the gradient of the interpolation of $\bar{\rho}\tilde{D}_T$. A detailed description of the functional form of MLG, the process for its evaluation, and numerical tests of its convergence properties are given in Appendix C.

6.3 PMC Performance of PPERM/MLG in an LES/PDF Context

In this subsection, we compare the PMC performance of the combination of PPERM and MLG interpolation schemes with that of standard multilinear and fourth-order Lagrangian interpolation. First we consider simulations without explicit PMC correction. Tables 4 and 5 present results from two types of simulations, respectively an FC simulation with $n_{pc} = 30$ and a flamelet model simulation with $n_{pc} = 50$. As mentioned previously, the SDE integration scheme used is the second-order KP scheme.

Simulation	FC, $n_{pc} = 30$, multilinear	FC, $n_{pc} = 30$, PPERM/MLG	FC, $n_{pc} = 30$, 4^{th} order
$\|\epsilon^c\|_1$	1.16×10^{-1}	7.92×10^{-2}	9.67×10^{-2}
$\|\epsilon^c\|_\infty$	3.95×10^{-1}	2.54×10^{-1}	3.17×10^{-1}
$\mu s/ptcl \cdot step$	84.2	91.1	102.3

Table 4: Comparison of the performance of different interpolation schemes for an FC simulation with $n_{pc} = 30$, without PMC correction.

As we can see on both tables, using fourth-order accurate Lagrangian interpolation provides an advantage over standard multilinear interpolation, and PPERM/MLG provides an even greater advantage - L^1 consistency errors are decreased by approximately 30%, and L^∞ consistency errors are decreased by approximately 35%, relative to the simulation with multilinear interpolation. Furthermore, the added computational cost of using PPERM/MLG is lower than that of using fourth-order Lagrangian interpolation, and lower than 10% of the overall computational cost for a simulation with full chemistry modeling using the FC mechanism. For flamelet calculations, the added computational cost of using the higher-order interpolation schemes is considerable, but the FC test case is more representative of a typical LES/PDF simulation.

Note that, even with fourth-order Lagrangian and PPERM/MLG interpolation, we still need to perform explicit PMC correction in order to meet the target values for ϵ_0^c - 0.171 and 0.132 for $n_{pc} = 30$ and $n_{pc} = 50$, respectively. For this

Simulation	Flamelet, $n_{pc} = 50$, multilinear	Flamelet, $n_{pc} = 50$, PPERM/MLG	Flamelet, $n_{pc} = 50$, 4^{th} order
$\|\epsilon^c\|_1$	1.03×10^{-1}	6.77×10^{-2}	8.91×10^{-2}
$\|\epsilon^c\|_\infty$	2.73×10^{-1}	1.88×10^{-1}	2.27×10^{-1}
$\mu s/ptcl \cdot step$	13.3	19.5	29.2

Table 5: Comparison of the performance of different interpolation schemes for an flamelet simulation with $n_{pc} = 50$, without PMC correction.

reason, we perform another set of simulations with the optimal explicit corrective scheme from section 5: $F(\epsilon^c) = F_2^{\alpha=3.0}(\epsilon^c)$, with $\tau^c = 4\Delta t$ and $\tau^c = 6\Delta t$ for $n_{pc} = 30$ and $n_{pc} = 50$, respectively - since this choice of parameters yielded $\|\epsilon^c\|_\infty < \epsilon_0^c$ in section 5, where multilinear interpolation was used, it is to be expected that for PPERM/MLG and fourth-order Lagrangian interpolation the resulting consistency errors will be even lower.

Tables 6 and 7 present the simulation results, again for FC simulations with $n_{pc} = 30$ and flamelet simulations with $n_{pc} = 50$. Time histories of the L^1 and L^∞ norms of the corrective velocities are also shown on fig. 9, on which we can see that while there is some fluctuation of the corrective velocity magnitudes over the simulation time span, the overall magnitude of the fields does not change considerably (the fluctuation is on the order of 20% of the mean value). Finally, fig. 10 displays instantaneous contour plots of the consistency error and axial component of the corrective velocity at the end of the FC, PPERM/MLG simulation. We can see on this figure that the PMC error field has length scales much smaller than those of the corrective velocity, which is to be expected due to the smoothing in the PMC corrective scheme.

Simulation	FC, $n_{pc} = 30$, multilinear	FC, $n_{pc} = 30$, PPERM/MLG	FC, $n_{pc} = 30$, 4^{th} order
$\|\epsilon^c\|_1$	4.8×10^{-2}	4.0×10^{-2}	4.3×10^{-2}
$\|\epsilon^c\|_\infty$	1.55×10^{-1}	1.20×10^{-1}	1.33×10^{-1}
$\|\epsilon^U\|_1$	4.7×10^{-2}	2.6×10^{-2}	3.6×10^{-2}
$\|\epsilon^U\|_\infty$	9.6×10^{-2}	4.2×10^{-2}	6.5×10^{-2}

Table 6: Comparison of the performance of different interpolation schemes for an FC simulation with $n_{pc} = 30$, with PMC correction.

As we can see, with the addition of PMC correction, the PPERM/MLG simulations again give a reduction in the consistency errors, though not as great as in Tables 4 and 5 - for example, PPERM/MLG interpolation yields a reduction of approximately 20% in $\|\epsilon^c\|_\infty$ over multilinear interpolation, compared

with the 35% reduction obtained previously. However, we also see a considerable decrease in corrective velocities when PPERM/MLG and fourth-order Lagrangian interpolation are used - based on the L^1 measures of corrective velocity, PPERM/MLG reduces the corrective velocities by approximately 45%, and fourth-order Lagrangian interpolation reduces them by approximately 25%.

Simulation	Flamelet, $n_{pc} = 50$, multilinear	Flamelet, $n_{pc} = 50$, PPERM/MLG	Flamelet, $n_{pc} = 50$, 4^{th} order
$\|\epsilon^c\|_1$	3.7×10^{-2}	2.9×10^{-1}	3.2×10^{-2}
$\|\epsilon^c\|_\infty$	1.31×10^{-1}	1.06×10^{-1}	1.09×10^{-1}
$\left\|\epsilon^U\right\|_1$	3.8×10^{-2}	2.2×10^{-2}	2.9×10^{-2}
$\left\|\epsilon^U\right\|_\infty$	7.9×10^{-2}	3.8×10^{-2}	6.0×10^{-2}

Table 7: Comparison of the performance of different interpolation schemes for an flamelet simulation with $n_{pc} = 50$, with PMC correction.

Based on these tests, we conclude that, from a PMC standpoint, the PPERM/MLG interpolation schemes have an advantage over both multilinear and fourth-order Lagrangian interpolation. Moreover, we note that the overall performance of the PPERM/MLG schemes with KP SDE integration, and PMC correction enforced by the optimal scheme from section 5 is quite satisfactory - the desired PMC criteria are achieved with corrective velocities of approximately 2.5% relative magnitude in the L^1 sense, and approximately 4% relative magnitude in the L^∞ sense.

7 Conclusions

This paper addresses the issue of particle mass consistency (PMC) in hybrid LES/PDF methods for turbulent reactive flows. The particle mass consistency error is defined as the departure from one of the ratio between particle mass density and LES resolved density. A set of new implicit and explicit PMC preservation and correction schemes is introduced. A second-order SDE integration scheme, in itself an implicit PMC preservation scheme when compared to Euler SDE integration, is shown to decrease the instantaneous PMC error of the simulation by 30% and the time-averaged, deterministic PMC error by 45%. Explicit PMC correction is optimized by employing a smoothed version of the consistency error field; compared to a standard explicit PMC scheme, the variant using the smoothed consistency error field decreases the corrective velocity necessary for attaining the desired accuracy criterion by 35% in the L^∞ sense and by 50% in the L^1 sense. The implicit schemes also include new velocity and scalar interpolation algorithms designed for this issue, which yield an

overall reduction of up to 45% in the corrective velocity necessary for maintaining PMC, relative to standard multilinear interpolation, and which moreover perform better than the standard 4^{th} order Lagrangian interpolation scheme. Using the optimal explicit and implicit schemes, the overall performance of the simulation is shown to be quite satisfactory, meeting the desired consistency criterion while using a corrective velocity of relatively low magnitude.

Acknowledgements

This work was supported in part by NSF Award No. CBET-1033246 and in part by Grant Number FA9550-09-1-0611 funded by the National Center for Hypersonic Combined Cycle Propulsion, sponsored by the AFOSR and NASA ARMD.

References

[1] P. Colucci, F. Jaberi, P. Givi, and S. B. Pope, *Filtered Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, Phys. Fluids **10** (1998), pp.499-515

[2] R.O. Fox *Computational Models for Turbulent Reactive Flows*, Cambridge University Press (2003)

[3] Y. Ge, M. J. Cleary, and A. Y. Klimenko, *Sparse-Lagrangian FDF Simulations of Sandia Flame E with Density Coupling*, Proc. Comb. Inst., **33**, (2011)

[4] D. C. Haworth, *Progress in Probability Density Function Methods for Turbulent Reacting Flows*, Prog. Energy Comb. Sci., **36**, 2, (2010), pp. 168-259

[5] F. Jaberi, P. Colucci, S. James, P. Givi, and S. B. Pope, *Filtered Mass Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, J. Fluid Mech. **401** (1999), pp.85-121

[6] P. Jenny, S. B. Pope, M. Muradoglu, and D. A. Caughey, *A Hybrid Algorithm for the Joint PDF Equation of Turbulent Reactive Flows*, J. Comput. Phys., **166** (2001), pp. 218-252

[7] P. E. Kloeden, and E. Platen *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, Berlin, 1992

[8] A. R. Masri, and R. W. Bilger, *Turbulent Diffusion Flames of Hydrocarbon Fuels Stabilized on a Bluff Body*, Proc. Comb. Inst., **20**, (1985)

[9] R. McDermott, and S.B. Pope, *A Particle Formulation for Treating Differential Diffusion in Filtered Density Function Methods*, J. Comput. Phys., **226** (2007), pp. 947-993

[10] R. McDermott, and S.B. Pope, *The Parabolic Edge Reconstruction Method (PERM) for Lagrangian Particle Advection*, J. Comput. Phys., **227** (2008), pp. 5447-5491

[11] M. Muradoglu, P. Jenny, S. B. Pope, and D. A. Caughey, *A Consistent Hybrid Finite-Volume/Particle Method for the PDF Equations of Turbulent Reactive Flows*, J. Comput. Phys., **154** (1999), pp. 342-371

[12] M. Muradoglu, S. B. Pope, and D. A. Caughey, *The Hybrid Method for the PDF Equations of Turbulent Reactive Flows: Consistency Conditions and Correction Algorithms*, J. Comput. Phys., **172** (2001), pp. 841-878

[13] R. Mustata, L. Valino, C. Jimenez, W. P. Jones, S. Bondi *A Probability Density Function Eulerian Monte Carlo Field Method for Large Eddy Simulations: Applications to a Turbulent Piloted Methane/Air Diffusion Flame (Sandia D)*, Combust. Flame, **145**, (2006), pp. 88-104

[14] C. D. Pierce, and P. Moin, *Progress-Variable Approach for Large-Eddy Simulation of Turbulent Combustion*, J. Fluid Mech.**504**, (2004), pp. 73-97

[15] H. Pitsch *Large-Eddy Simulation of Turbulent Combustion*, Annu. Rev. Fluid Mech., **38**, (2006), pp. 453-482

[16] S. B. Pope *A Monte Carlo Method for the PDF Equations of Turbulent Reactive Flow*, Combust. Sci. Tech., **25**, (1981), pp. 159-174

[17] S. B. Pope, *PDF Methods for Turbulent Reactive Flows*, Prog. Energy Combust. Sci., **11**, pp. 119-192 (1985)

[18] S. B. Pope *Computations of Turbulent Combustion: Progress and Challenges*, Proc. Comb. Inst., **23**, (1990), pp. 591-612

[19] S. B. Pope, *Computationally Efficient Implementation of Combustion Chemistry Using In Situ Adaptive Tabulation*, Comb. Theory Modelling **1** (1997), pp.41-63

[20] S. B. Pope, *Turbulent Flows*, Cambridge University Press, Cambridge, 2000

[21] S.B. Pope, *Self-Conditioned Fields for Large-Eddy Simulations of Turbulent Flows*, J. Fluid Mech., **652** (2010), pp. 139-169

[22] P.P. Popov, H. Wang, and S.B. Pope, *Specific Volume Coupling and Convergence Properties in Hybrid Particle/Finite Volume Algorithms for Turbulent Reactive Flows*, J. Comput. Phys., submitted

[23] V. Raman, and H. Pitsch, *A Consistent LES/Filtered Density Function Formulation for the Simulation of Turbulent Flames with Detailed Chemistry*, Proc. Comb. Inst., **31**, (2007), pp. 1711-1719

[24] L. Valiño *A Field Monte Carlo Formulation for Calculating the Probability Density Function of a Single Scalar in a Turbulent Flow*, Flow Turb. Combust., **60**, (1998), pp. 157-172

[25] J. Villermaux, and J.C. Devillon *Reprèsentation de la redistribution des domaines de sègrègation dans un fluide par un modèle d'interaction phènomènologique*, 2nd Int. Symp. Chem. React. Engng, Amsterdam, 1972

[26] S. Viswanathan, H. Wang, and S. B. Pope, *Numerical Implementation of Mixing and Molecular Transport in LES/PDF Studies of Turbulent Reacting Flows*, J. Comput. Phys. **230**, (2011), pp. 6916-6957

[27] H. Wang, and S. B. Pope *Large Eddy Simulation/Probability Density Function Modeling of a Turbulent $CH_4/H_2/N_2$ Jet Flame*, Proc. Comb. Inst., **33**, (2011), pp. 1319-1330

[28] Y. Z. Zhang, and D. C. Haworth, *A General Particle Mass Consistency Algorithm for Hybrid Particle/Finite Volume PDF Methods*, J. Comp. Phys., **194**, (2004), pp. 156-193

Appendix A: A Model for the Development of Mass-Consistency Errors in the Shear Layer

In this section of the Appendix, we describe a simple 2D analytic test case which reproduces the deterministic (time-averaged) PMC errors observed in section 4. Particles of equal mass are initialized uniformly on the domain $x \in [0,1] \times y \in [0,1]$, with a density of 10^6 particles per unit area, and evolve by the SDE

$$dX^* = \frac{\partial D_T}{\partial x}\left(X^*\right) dt + D_T\left(X^*\right) dW_1^* \tag{39}$$

$$dY^* = V\left(X^*\right) dt + D_T\left(X^*\right) dW_2^*, \tag{40}$$

for

$$V(x) = 3\left(0.1 + erf((x-0.5)/0.05) - erf((x-0.85)/0.05)\right) \tag{41}$$

and

$$D_T(x) = 0.02 + exp\left(-((x-0.5)/0.05)^2\right) \tag{42}$$

which, from eq.(17) implies that $\frac{D^*q}{Dt^*} = 0$, hence in the absence of numerical errors the particle distribution should remain uniform. The velocity and diffusivity fields defined in eqs.(41,42), which are plotted on fig. 7, are qualitatively similar to the velocity and drift fields in the bluff-body flame: there is a shear layer at $x = 0.5$, in which region the diffusivity is considerably higher than everywhere else in the domain. Another shear layer is located at $x = 0.85$, for the sake of having the fields be approximately periodic in x (with a negligible discontinuity at $x = 1$), but we shall see that this shear layer does not influence the particle distribution as much, because it is located in a region of low diffusivity.

All particles are advected by either the Euler or KP SDE time integration schemes, with a time step of $\Delta t = 0.0125 \times p$, for $400/p$ time steps, where $p = 1, 1/2, 1/4, 1/8$ is a time step refinement parameter. A periodic boundary condition in the $x-$direction is enforced. In the $y-$direction, we impose inflow boundary conditions by deleting, after each time step, all particles in the region $y \in [0, 0.3] \cup [0.7, 1.0]$, and initializing new particles in that region, with a uniform density of 10^6 particles per unit area. Thus, the inlets are the lines $y = 0.3$ and $y = 0.7$.

The results for $p = 1$ are plotted on the right-hand side of fig. 7. The figure shows values of ϵ^c, averaged over time and over the $y-$coordinate, from $y = 0.3$ to $y = 0.45$, plotted against x. We see a region of increased particle density (positive ϵ^c) at the low-velocity part of the shear layer, $0.0 < x < 0.4$, and a region of decreased particle density at the high-velocity part of the shear layer, $0.4 < 0 < 0.7$. Qualitatively, this agrees well with the deterministic PMC error fields observed in section 4. Figure 8 plots values of $max(\epsilon^c)$ against the time

step. As can be expected, we see that $max(\epsilon^c)$ converges in a second-order fashion for the KP scheme, and in a first-order fashion for the Euler scheme.

This test case demonstrates that considerable deterministic PMC errors can develop even when the only source of numerical error is the error due to the SDE integration scheme. It also illustrates the advantage of using a higher-order SDE integration scheme: even for the largest time steps, the PMC error is reduced by a factor of 3 when the second-order KP scheme is used instead of the Euler scheme.

Appendix B: The PPERM Velocity Interpolation Scheme

In this section of the appendix, we describe in detail the functional form of the PPERM velocity interpolation scheme, and the algorithm for its evaluation. Consider a given cell $x \in (x_0, x_1)$, $r \in (r_0, r_1)$, $\theta \in (\theta_0, \theta_1)$ in cylindrical coordinates, and denote $\Delta x = x_1 - x_0$, $\Delta r = r_1 - r_0$, $\Delta\theta = \theta_1 - \theta_0$.

Here, we use the subscript index 1 to denote a positive cell face (e.g., $\{(x, r, \theta) \,|x = x_1\}$) and the subscript index 0 to denote a negative cell face (e.g., $\{(x, r, \theta) \,|x = x_0\}$). We shall also use subscript index pairs to denote cell edges (e.g., for a radial edge, the subscript index 01 denotes $\{(x, r, \theta) \,|x = x_0, \theta = \theta_1\}$), and subscript index triples to denote cell vertices (e.g., for a radial edge, the subscript index 101 denotes $\{x = x_1, r = r_0, \theta = \theta_1\}$): in addition to the 8 cell vertices, we shall also use the subscripts $i\frac{1}{2}k$, for $i, k \in \{0, 1\}$ to denote the midpoints of the four radial cell edges. This is illustrated on fig. 13.

Introducing the local coordinates

$$a_1 = \frac{x - x_0}{x_1 - x_0}, b_1 = \frac{r - r_0}{r_1 - r_0}, c_1 = \frac{\theta - \theta_0}{\theta_1 - \theta_0},$$

$$a_0 = 1 - a_1, b_0 = 1 - b_1, c_0 = 1 - c_1, \tag{43}$$

and the coefficients $U_{jk}^{[l]}, V_{ik}^{[m]}, W_{ij}^{[n]}$, for $i, j, k \in \{0, 1\}$, $l, n \in \{0, 1, 2\}$, and $m \in \{0, 1, 2, 3\}$ (where the superscripts correspond to the degree of the respective polynomial term), we define the PPERM-interpolated velocity $I(\mathbf{u})(x, r, \theta)$ in cylindrical coordinates as

$$I(\mathbf{u})_x = \sum_{j,k=0}^{1} b_j c_k \left(U_{jk}^{[0]} + (a_1 - 0.5) U_{jk}^{[1]} + \left((a_1 - 0.5)^2 - 0.25 \right) U_{jk}^{[2]} \right) \tag{44}$$

$$rI(\mathbf{u})_r = \sum_{i,k=0}^{1} a_i c_k (V_{jk}^{[0]} + (b_1 - 0.5) V_{jk}^{[1]} + \left((b_1 - 0.5)^2 - 0.25 \right) V_{jk}^{[2]}$$

$$+ b_1 (b_1 - 0.5)(b_1 - 1) V_{jk}^{[3]} \Big) \tag{45}$$

$$I(\mathbf{u})_\theta = \sum_{i,j=0}^{1} a_i b_j \left(W_{ij}^{[0]} + (c_1 - 0.5) W_{ij}^{[1]} + \left((c_1 - 0.5)^2 - 0.25 \right) W_{ij}^{[2]} \right). \quad (46)$$

Note that instead of interpolating for u_r, we interpolate for ru_r – this is done so that the divergence formula, which has the cylindrical coordinate form

$$r(\nabla \cdot \mathbf{u}) = r\frac{\partial u_x}{\partial x} + \frac{\partial r u_r}{\partial r} + \frac{\partial u_\theta}{\partial \theta}, \quad (47)$$

yields the following result for the interpolated divergence:

$$r(\nabla \cdot I(\mathbf{u})) = r \sum_{j,k=0}^{1} b_j c_k \left(U_{jk}^{[1]} + 2(a_1 - 0.5) U_{jk}^{[2]} \right) / \Delta x$$

$$+ \sum_{i,k=0}^{1} a_i c_k \left(V_{jk}^{[1]} + 2(b_1 - 0.5) V_{jk}^{[2]} + \left(3(b_1)^2 - 3b_1 + 0.5 \right) V_{jk}^{[3]} \right) / \Delta r$$

$$+ \sum_{i,j=0}^{1} a_i b_j \left(W_{ij}^{[1]} + (2c_1 - 0.5) W_{ij}^{[2]} \right) / \Delta \theta. \quad (48)$$

Also note that we are using an additional 3^{rd} degree polynomial in the interpolation of ru_r. We do this so that we can get $r(\nabla \cdot I(\mathbf{u}))$ to be quadratic in b and linear in a, c - this enables us to obtain 3^{rd} order accurate values of $r(\nabla \cdot I(\mathbf{u}))$ near the centerline (i.e., in the limit as $r \downarrow 0$), which is needed for 2^{nd} order accurate values of $r(\nabla \cdot I(\mathbf{u}))$ near the centerline.

Next, we describe the algorithm for the evaluation of the coefficients $U_{jk}^{[l]}, V_{ik}^{[m]}, W_{ij}^{[n]}$.

1. **Interpolation of FV velocity and divergence onto cell corners and radial edge midpoints:** Using standard linear interpolation, evaluate second-order accurate approximations of u_x, u_r, u_θ at the corners of the present cell and its neighbors: denote these as $u_{x,ijk}, u_{r,ijk}, u_{\theta,ijk}$ respectively, for $i, j, k \in \{-1, 0, 1, 2\}$. Also, evaluate second-order accurate approximations of $(\nabla \cdot \mathbf{u})$ at the 8 cell corners and 4 radial edge midpoints of the present cell: denote these as d_{ijk}, for $i, k \in \{0, 1\}$, $j \in \left\{0, \frac{1}{2}, 1\right\}$.

2. **Get provisional values for the 0^{th} and 1^{st} order PPERM coefficients:** Set

$$U_{jk}^{[1]} = u_{x,1jk} - u_{x,0jk}, U_{jk}^{[1]} = (u_{x,1jk} + u_{x,0jk})/2, \quad (49)$$

$$V_{ik}^{[1]} = r_1 u_{r,i1k} - r_0 u_{r,i0k}, V_{ik}^{[1]} = \left(r_1 u_{r,i1k} + r_0 u_{r,i0k} \right)/2, \quad (50)$$

$$W_{ij}^{[1]} = u_{\theta,ij1} - u_{\theta,ij0}, W_{jk}^{[1]} = (u_{\theta,ij1} + u_{\theta,ij0})/2. \quad (51)$$

Since the second and third degree PPERM terms vanish at the cell corners, eqs.(7-9) ensure that at this point in the interpolation process the corner values of u_x, u_r, u_θ are exactly equal to the second-order accurate approximations from Step 1.

3. **Flux correction for the 0^{th} and 1^{st} order PPERM coefficients:** For the 6 cell faces, subtract the face-averages of the interpolated velocity, implied by the coefficients obtained in Step 2, from the FV face-average velocities. Denote these differences respectively $\Delta u_{x,1}$ and $\Delta u_{x,0}$ for the positive and negative $x-$faces, $\Delta u_{r,1}$ and $\Delta u_{r,0}$ for the positive and negative $r-$faces, and $\Delta u_{\theta,1}$ and $\Delta u_{\theta,0}$ for the positive and negative $\theta-$face. Using $\tilde{u}_{x,i}, \tilde{u}_{r,i}, \tilde{u}_{\theta,i}$, for $\imath \in \{0,1\}$ to denote the FV face-average, this calculation takes the form

$$\lambda_1 = \frac{(1/2)\, 2r_0 + (1/6)\,(r_1 - r_0)}{r_0 + r_1}, \lambda_0 = 1 - \lambda_1 \tag{52}$$

$$\Delta u_{x,i} = \tilde{u}_{x,i} - \sum_{j,k=0}^{1} \lambda_j \left(U_{jk}^{[0]} + (i - 0.5)\, U_{jk}^{[1]} \right) /2 \tag{53}$$

$$\Delta u_{r,j} = \tilde{u}_{r,j} - \sum_{i,k=0}^{1} \left(V_{ik}^{[0]} + (j - 0.5)\, V_{ik}^{[1]} \right) /(4r_j) \tag{54}$$

$$\Delta u_{\theta,k} = \tilde{u}_{\theta,k} - \sum_{i,j=0}^{1} \left(W_{ij}^{[0]} + (k - 0.5)\, W_{ij}^{[1]} \right) /4 \tag{55}$$

Note that this calculation can be performed at this stage, as the velocity flux through the cell faces depends only on the 0^{th} and 1^{st} order PPERM coefficients. Next, perform the corrections:

$$U_{jk}^{[0]} \leftarrow U_{jk}^{[0]} + \left(\Delta u_{x,1} + \Delta u_{x,0}\right)/2, U_{jk}^{[1]} = U_{jk}^{[1]} + \left(\Delta u_{x,1} - \Delta u_{x,0}\right), \tag{56}$$

$$V_{ik}^{[0]} \leftarrow V_{ik}^{[0]} + \left(r_1 \Delta u_{r,1} + r_0 \Delta u_{r,0}\right)/2, V_{ik}^{[1]} = V_{ik}^{[1]} + \left(r_1 \Delta u_{r,1} - r_0 \Delta u_{r,0}\right), \tag{57}$$

$$W_{ij}^{[0]} \leftarrow W_{ij}^{[0]} + \left(\Delta u_{\theta,1} + \Delta u_{\theta,0}\right)/2, W_{ij}^{[1]} = W_{ij}^{[1]} + \left(\Delta u_{\theta,1} - \Delta u_{\theta,0}\right). \tag{58}$$

This ensures that the face averages of the interpolated velocity are identically equal to the FV face-average velocities, and hence that the total flux of interpolated velocity through the cell is equal to the FV cell flux.

4. **First estimate for the 2^{nd} and 3^{rd} order PPERM coefficients.** From the cell corner values of velocity obtained in Step 1, calculate 4-point approximations of the second derivatives of u_x, ru_r, u_θ along the cell edges parallel to the respective velocity component, and set $U_{jk}^{[2]}, V_{ik}^{[2]}, W_{ij}^{[2]}$ so that the edge-averages of second derivatives of the interpolated velocity

match the approximated FV second derivatives. For a uniform grid, this calculation reduces to

$$U_{jk}^{[2]} = \frac{1}{4}\left(u_{x,2jk} - u_{x,1jk} - u_{x,0jk} + u_{x,-1jk}\right) \tag{59}$$

$$V_{ik}^{[2]} = \frac{1}{4}\left(\left(r_1 + \Delta r\right) u_{r,i2k} - r_1 u_{r,i1k} - r_0 u_{r,i0k} + \left(r_0 - \Delta r\right) u_{r,i-1k}\right) \tag{60}$$

$$W_{ij}^{[2]} = \frac{1}{4}\left(u_{\theta,ij2} - u_{\theta,ij1} - u_{\theta,ij0} + u_{\theta,ij-1}\right). \tag{61}$$

Next, calculate 4-point approximations of the third derivatives of ru_r along the radial cell edges, and set $V_{ik}^{[3]}$ so that the edge-averages of third derivatives of the interpolated velocity match the approximated FV second derivatives. For a uniform grid, this calculation reduces to

$$V_{ik}^{[3]} = \frac{1}{6}\left(\left(r_1 + \Delta r\right) u_{r,i2k} - 3r_1 u_{r,i1k} + 3r_0 u_{r,i0k} - \left(r_0 - \Delta r\right) u_{r,i-1k}\right). \tag{62}$$

5. **Divergence correction for the 2^{nd} and 3^{rd} order PPERM coefficients.** Subtract from the second-order approximations of the divergence, d_{ijk}, calculated in Step 1, the divergence of the interpolated velocity, with the present PPERM coefficients, denoted by $\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{ijk}$, at the respective locations. Denote the resulting differences as δd_{ijk}, for $i, j \in \{0, 1\}$, $j \in \left\{0, \frac{1}{2}, 1\right\}$. Let A be the 12×16 matrix (calculated from eqs.(44-46)) which relates a change in the 16-component vector $\left(U_{jk}^{[2]}, V_{ik}^{[2]}, V_{ik}^{[3]}, W_{ij}^{[2]}\right)$ to a change in the 12-component vector $\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{ijk}$:

$$\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{ijk} = A\Big(\delta U_{jk}^{[2]}, \delta V_{ik}^{[2]}, \delta V_{ik}^{[3]}, \delta W_{ij}^{[2]}\Big). \tag{63}$$

Correct the 2^{nd} and 3^{rd}-order PPERM coefficients by setting

$$U_{jk}^{[2]} \leftarrow U_{jk}^{[2]} + \delta U_{jk}^{[2]}, \tag{64}$$

$$V_{ik}^{[2]} \leftarrow V_{ik}^{[2]} + \delta V_{ik}^{[2]},\ V_{ik}^{[3]} = V_{ik}^{[3]} + \delta V_{ik}^{[3]}, \tag{65}$$

$$W_{ij}^{[2]} \leftarrow W_{ij}^{[2]} + \delta W_{ij}^{[2]}, \tag{66}$$

where the 16-component vector $\left(\delta U_{jk}^{[2]}, \delta V_{ik}^{[2]}, \delta V_{ik}^{[3]}, \delta W_{ij}^{[2]}\right)$ is the least squares-minimal norm (LSMN) solution of the linear system

$$A\Big(\delta U_{jk}^{[2]}, \delta V_{ik}^{[2]}, \delta V_{ik}^{[3]}, \delta W_{ij}^{[2]}\Big) = \delta d_{ijk}. \tag{67}$$

This concludes the PPERM evaluation procedure. We note that the 12×16 matrix A has rank 11, and its kernel corresponds to to the constraint

$$\sum_{i,k=0}^{1} \frac{1}{r_0}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i0k} + \frac{4}{r_0 + \Delta r/2}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i\frac{1}{2}k} + \frac{1}{r_1}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i1k} = 0, \tag{68}$$

or equivalently, to the fact that altering the 2^{nd} and 3^{rd} degree PPERM coefficients does not change the total velocity flux through the cell.
For a discretely divergence-free FV velocity field, the correction procedure of Step 3 ensures that the total flux through the cell is zero. Due to the fact that $r\left(\nabla \cdot I\left(\mathbf{u}\right)\right)$ is linear in a, c and quadratic in b by design (from eqs.(44-47)), this implies that

$$\sum_{i,k=0}^{1} \frac{1}{r_0}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i0k} + \frac{4}{r_0 + \Delta r/2}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i\frac{1}{2}k} + \frac{1}{r_1}\delta\left[\nabla \cdot I\left(\mathbf{u}\right)\right]_{i1k} = 0, \tag{69}$$

and so for this case the LSMN solution of eq.(67) is an exact solution. Therefore, for a discretely divergence-free FV velocity field, the divergence of the interpolated velocity is zero at the cell corners and the midpoints of the 4 radial cell edges. Again, since $r\left(\nabla \cdot I\left(\mathbf{u}\right)\right)$ is linear in a, c and quadratic in b, $r\left(\nabla \cdot I\left(\mathbf{u}\right)\right) = 0$ at the abovementioned locations implies that $r\left(\nabla \cdot I\left(\mathbf{u}\right)\right) = 0$ everywhere in the cell, which establishes the divergence-free property of the PPERM interpolated velocity, when applied to a discretely divergence-free FV velocity field.

A numerical test of the convergence properties of PPERM is performed on the cylindrical domain $x \in [0,1] \times r \in [0,1] \times \theta \in [0, 2\pi]$ discretized on an $N \times N \times 2N$ cylindrical grid, so that $\Delta x = 1/N$. An analytic velocity field is specified in Cartesian coordinates:

$$\begin{aligned} u_x(x,y,z) &= \sin\left(2\pi(x-0.57)\right)\cos\left(2\pi(y+0.23)\right) \\ u_y(x,y,z) &= -\cos\left(2\pi(x-0.57)\right)\sin\left(2\pi(y+0.23)\right) \\ u_z(x,y,z) &= -\cos\left(2\pi(z+x+0.57)\right)\sin\left(2\pi(y+z+0.23)\right) \end{aligned} \tag{70}$$

Figure 11 is a log-log plot of the L^∞ velocity and divergence errors, plotted against Δx. As we can see on the figure, both the velocity and divergence errors fall closely to a line of slope 2, which verified the second-order spatial convergence of the PPERM velocity and divergence.

To verify the divergence-free property for discretely divergence-free FV fields, we make the modification $u_z = 0$ in eq.(70), which yields a divergence-free velocity field. Then, for the $20 \times 20 \times 40$ cylindrical grid, the maximal divergence in the domain has magnitude 1.37×10^{-13}, which can be attributed to roundoff error.

Appendix C: The MLG Scalar Interpolation Scheme

In this appendix, we present the functional form of the MLG scalar interpolation scheme. Similarly to Appendix B, consider a given cell $x \in [x_0, x_1]$, $r \in [r_0, r_1]$, $\theta \in [\theta_0, \theta_1]$ in cylindrical coordinates, and denote $\Delta x = x_1 - x_0$, $\Delta r = r_1 - r_0$, $\Delta\theta = \theta_1 - \theta_0$. Define

$$x_C = \frac{x_0 + x_1}{2}, r_C = \frac{\left(r_0\right)^2 + \left(r_1\right)^2}{r_0 + r_1}, \theta_C = \frac{\theta_0 + \theta_1}{2}, \tag{71}$$

and set the local coordinates a, b, c so that

$$a = x - x_C, b = r - r_C, c = \theta - \theta_C. \tag{72}$$

Then, the MLG scalar interpolation of a scalar φ has the functional form of a truncated Taylor series in (x, r, θ) about the point (x_C, r_C, θ_C):

$$\begin{aligned} I(\varphi) = \Phi &+ abc\left(\Phi_{abc} + a\Phi_{a^2bc} + b\Phi_{ab^2c} + c\Phi_{abc^2}\right) \\ &+ a\left(\Phi_a + b\Phi_{ab} + a\Phi_{a^2} + ab\Phi_{a^2b} + b^2\Phi_{ab^2}\right) \\ &+ b\left(\Phi_b + c\Phi_{bc} + b\Phi_{b^2} + bc\Phi_{b^2c} + c^2\Phi_{bc^2}\right) \\ &+ c\left(\Phi_c + a\Phi_{ac} + c\Phi_{c^2} + ac\Phi_{ac^2} + a^2\Phi_{a^2c}\right), \end{aligned} \tag{73}$$

where the quantities Φ are scalar coefficients. Using cell-average FV values of φ from the current cell and its closest neighbors, we set Φ_0 to equal the cell average of φ in the current cell, which ensures that the cell average of the interpolated $I(\varphi)$ equals the FV cell average, and we determine all other Φ coefficients by numerically approximating the respective derivative of φ at (x_C, r_C, θ_C).
The truncated Taylor series of eq.(73) contains all first- and second- order terms of the full Taylor series expansion, and hence the interpolated $I(\varphi)$ is third-order accurate with respect to the grid spacing. Moreover,

$$\begin{aligned} \frac{\partial\left(I(\varphi)\right)}{\partial a} = &bc\left(\Phi_{abc} + 2a\Phi_{a^2bc} + b\Phi_{ab^2c} + c\Phi_{abc^2}\right) + \Phi_a + b\Phi_{ab} + a\Phi_{a^2} \\ &+ b^2\Phi_{ab^2} + 2ab\Phi_{a^2b} + c\Phi_{ac} + c^2\Phi_{ac^2} + 2ac\Phi_{a^2c}, \end{aligned} \tag{74}$$

which contains all first-order terms in the Taylor series expansion of $\frac{\partial(I(\varphi))}{\partial a}$, and so $\frac{\partial(I(\varphi))}{\partial a}$, and analogously $\frac{\partial(I(\varphi))}{\partial b}$ and $\frac{\partial(I(\varphi))}{\partial c}$, are second-order accurate with respect to the grid spacing.

Similarly to Appendix B, a numerical test of the convergence properties of MLG is performed on the cylindrical domain $x \in [0,1] \times r \in [0,1] \times \theta \in [0, 2\pi]$ discretized on an $N \times N \times 2N$ cylindrical grid, so that $\Delta x = 1/N$. The analytic scalar field field is specified in Cartesian coordinates:

$$\varphi(x, y, z) = 0.5 + 0.35 \sin\left(2\pi(x + y + z + t)\right) \cos\left(2\pi(x - y + z - t)\right) \tag{75}$$

Figure 12 is a log-log plot of the L^∞ scalar and scalar gradient errors, plotted against Δx. As we can see on the figure, the scalar errors fall closely to a line of slope 3, whereas the scalar gradient errors fall closely to a line of slope 2, which verifies MLG's convergence properties.

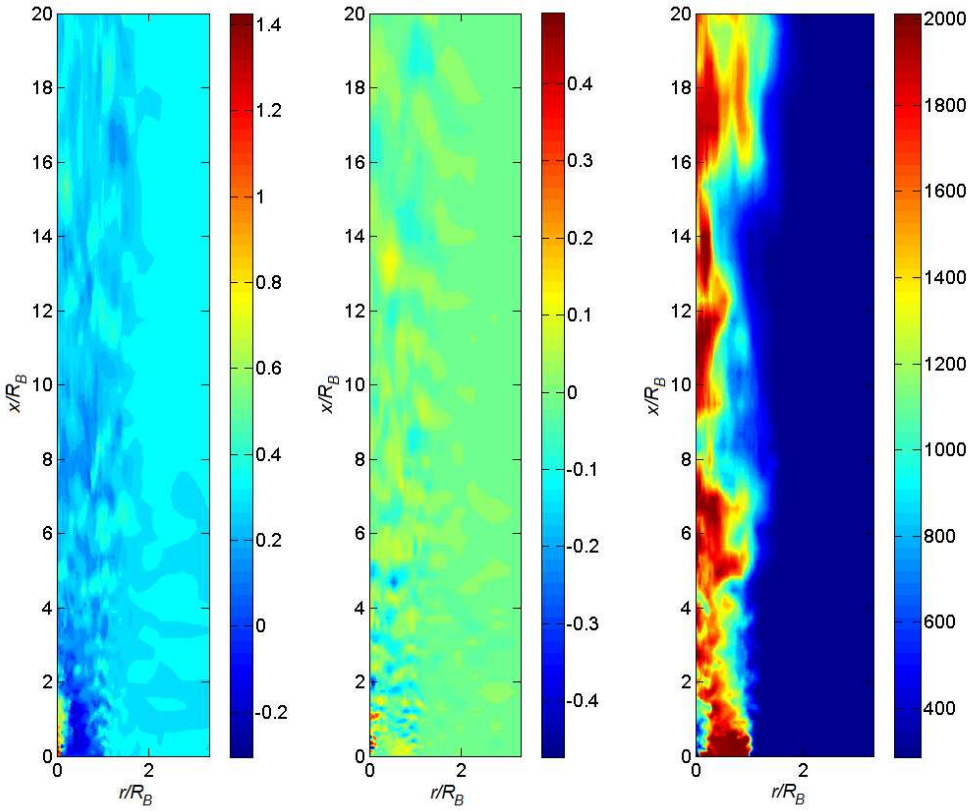

Figure 1: Instantaneous contour plots at the end of an FC simulation of the HM1 test case. Left: axial velocity, normalized by the jet bulk velocity. Middle: radial velocity, normalized by the jet bulk velocity. Right: Temperature (K).

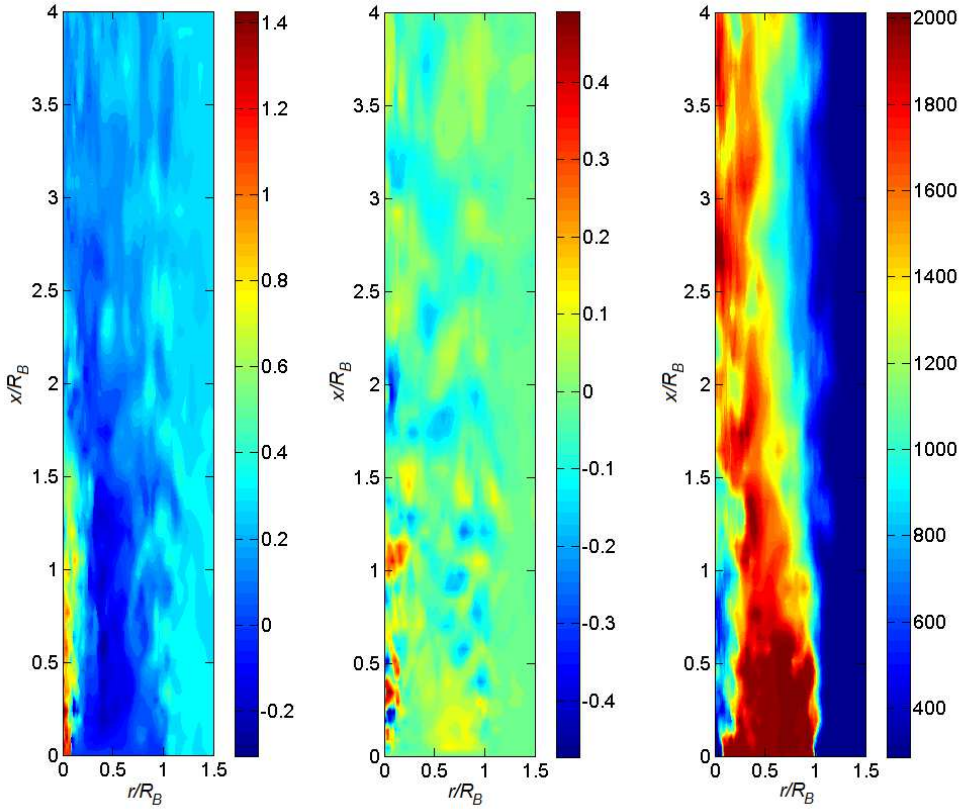

Figure 2: Instantaneous contour plots (enlarged view of the recirculation region) at the end of an FC simulation of the HM1 test case. Left: axial velocity, normalized by the jet bulk velocity. Middle: radial velocity, normalized by the jet bulk velocity. Right: Temperature (K).

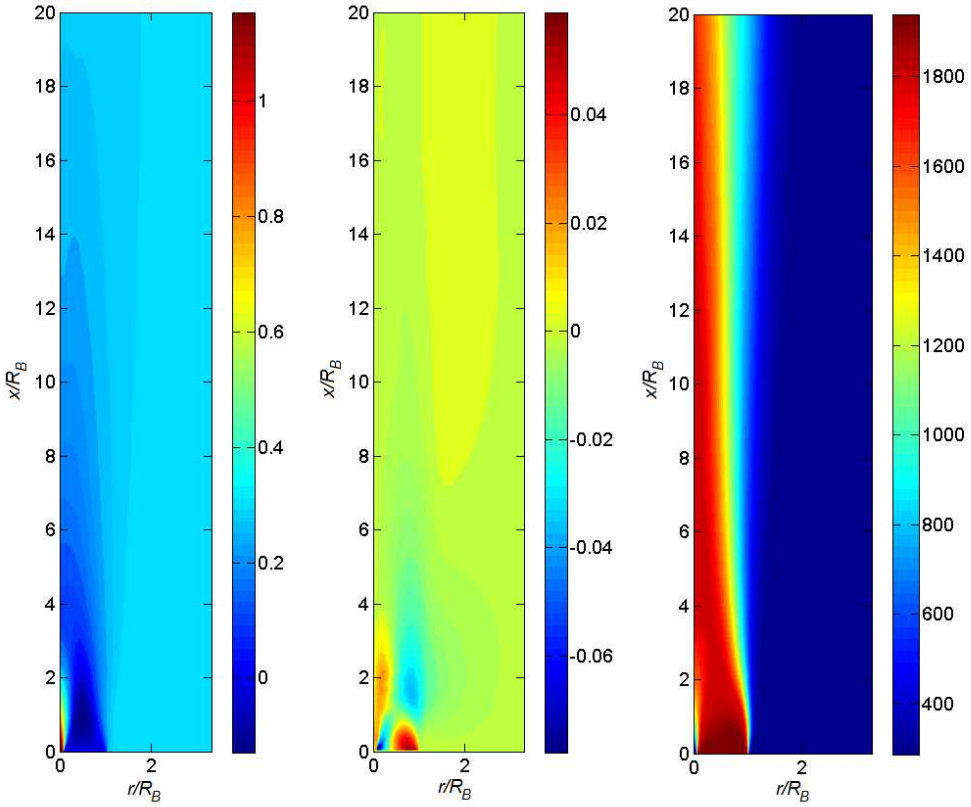

Figure 3: Contour plots of time-averaged fields in the FC simulation of the HM1 test case. Left: axial velocity, normalized by the jet bulk velocity. Middle: radial velocity, normalized by the jet bulk velocity. Right: Temperature (K).

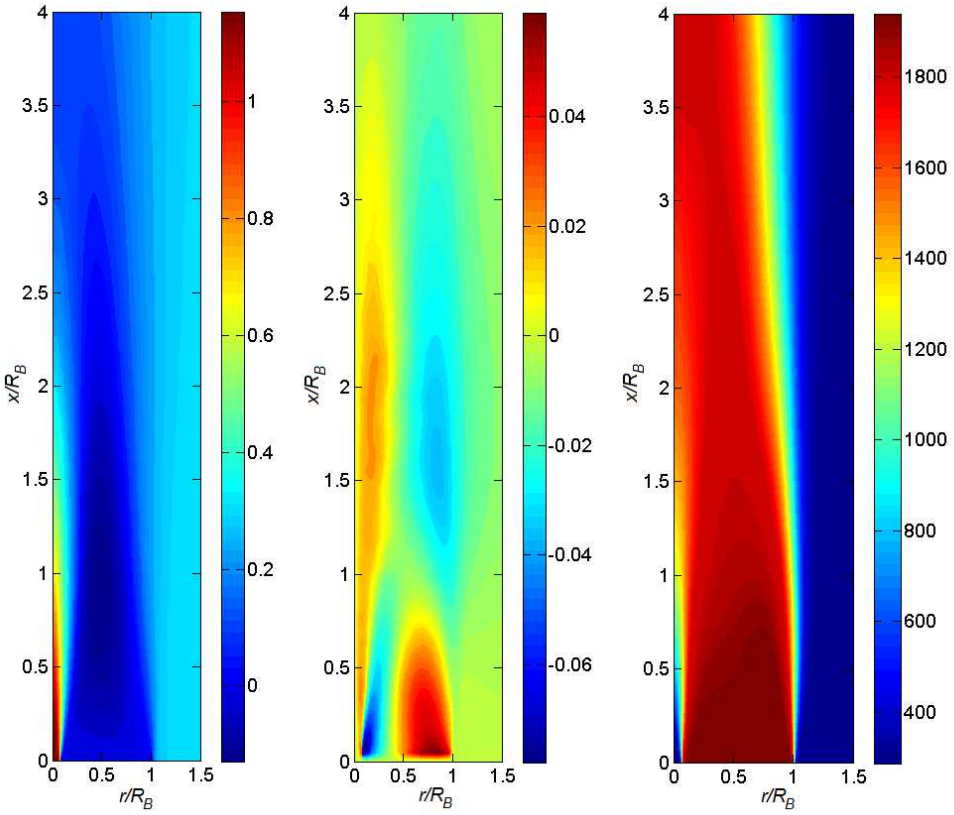

Figure 4: Contour plots (enlarged view of the recirculation region) of time-averaged fields in the FC simulation of the HM1 test case. Left: axial velocity, normalized by the jet bulk velocity. Middle: radial velocity, normalized by the jet bulk velocity. Right: Temperature (K).

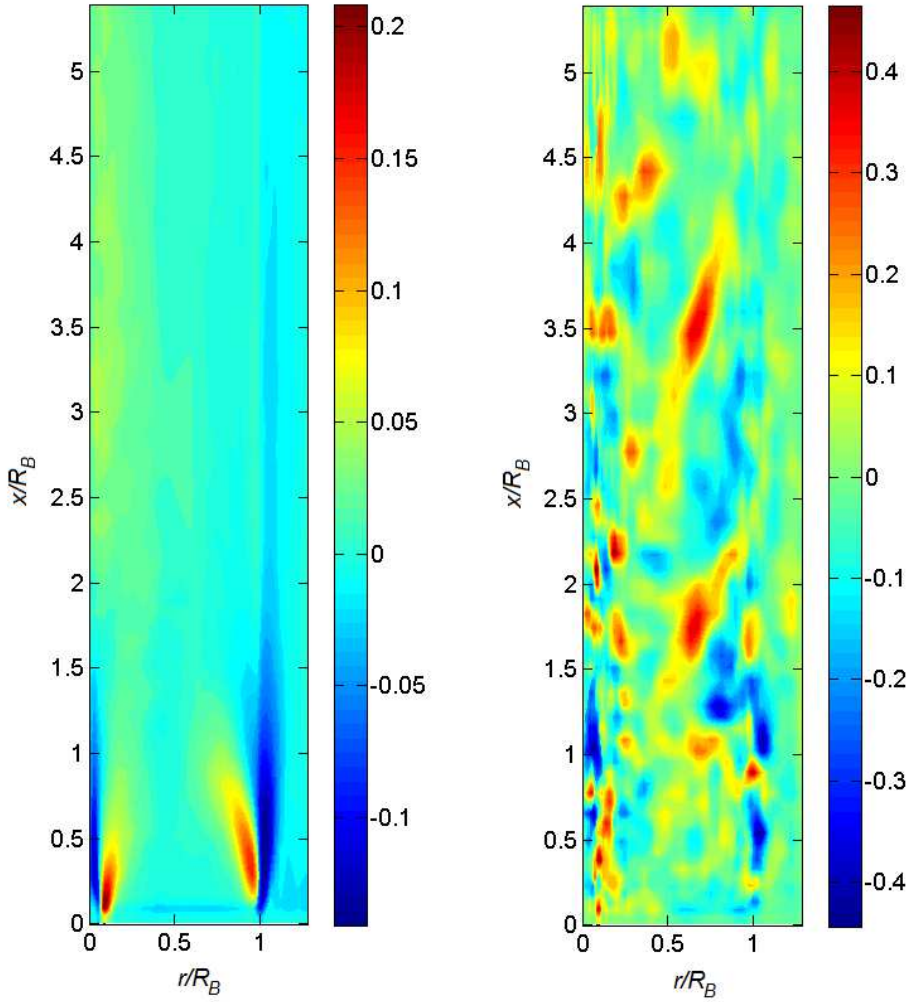

Figure 5: Contour plots of the PMC error, ϵ^c for an FC simulation with multilinear interpolation, $n_{pc} = 30$ and no explicit PMC correction. Left: contour plot of ϵ^c averaged in time over the second half of the simulation. Right: contour plot of ϵ^c at the last simulation time step.

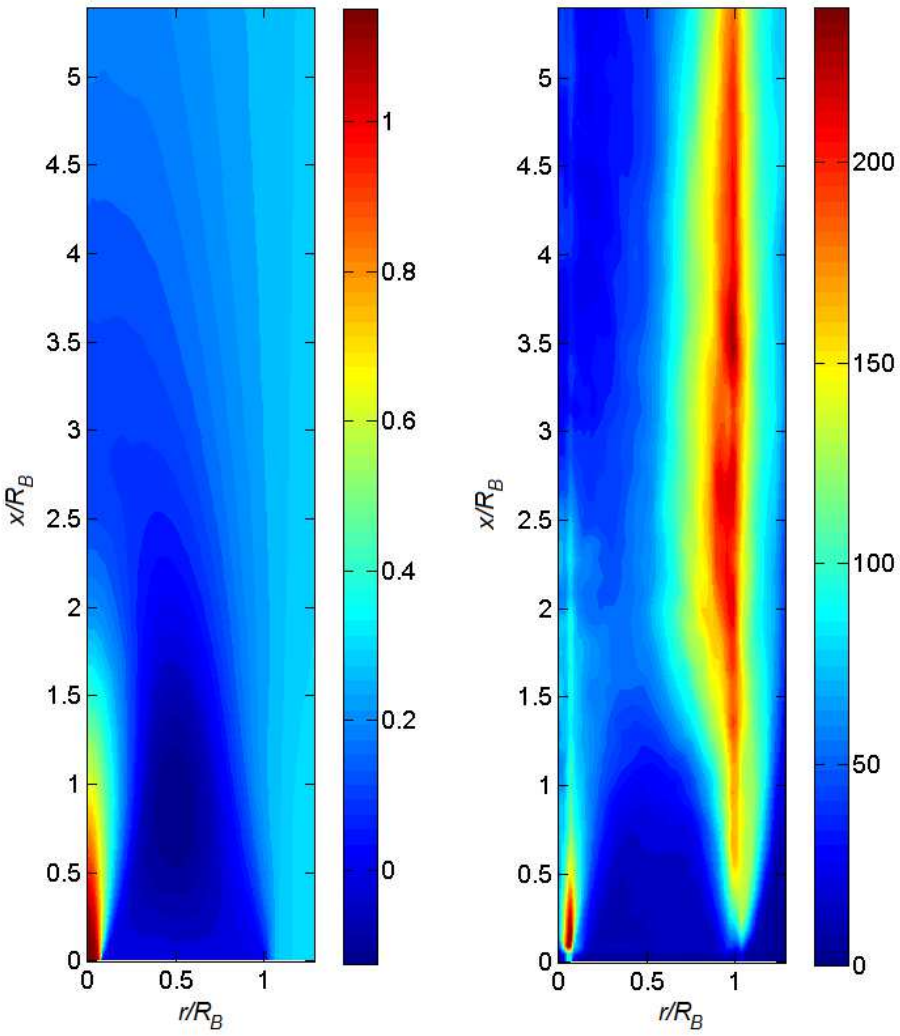

Figure 6: For the test case described in Appendix A, left: contour plot of $\langle \tilde{u}_x \rangle_t / U_J$, the time-averaged axial velocity normalized by the jet bulk velocity. Right: contour plot of $\frac{\langle \tilde{D}_T \rangle_t}{\Delta^2/\tau_J}$, the time-averaged turbulent diffusivity normalized by the LES filter size, Δ, and the jet characteristic time scale $\tau_J = 2R_B/U_J$.

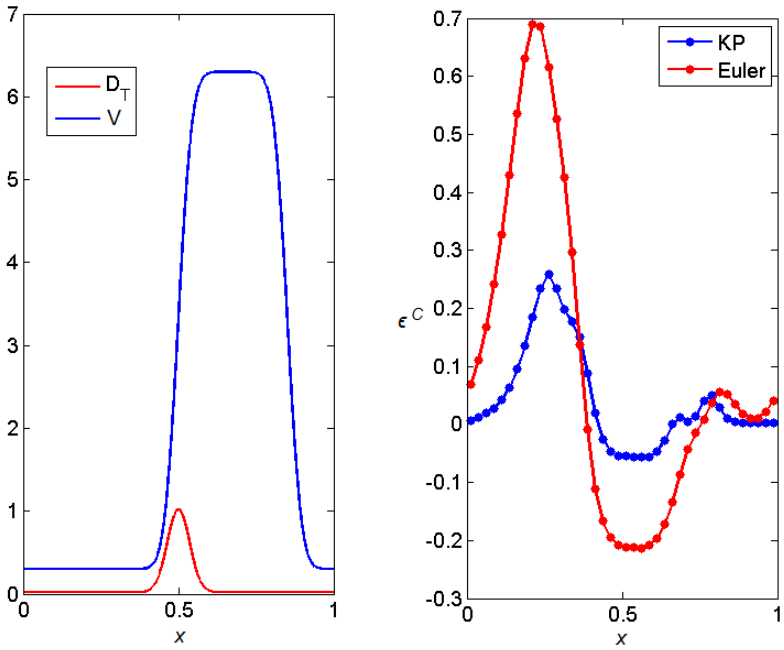

Figure 7: Results from the 2D PMC test case described in Appendix A. Left: plots of $V(x)$ and $D_T(x)$. Right: plots of time-averaged PMC error, $\langle \epsilon^c \rangle_T$ for the region $0.3 < y < 0.45$, and for $\Delta t = 0.0125$.

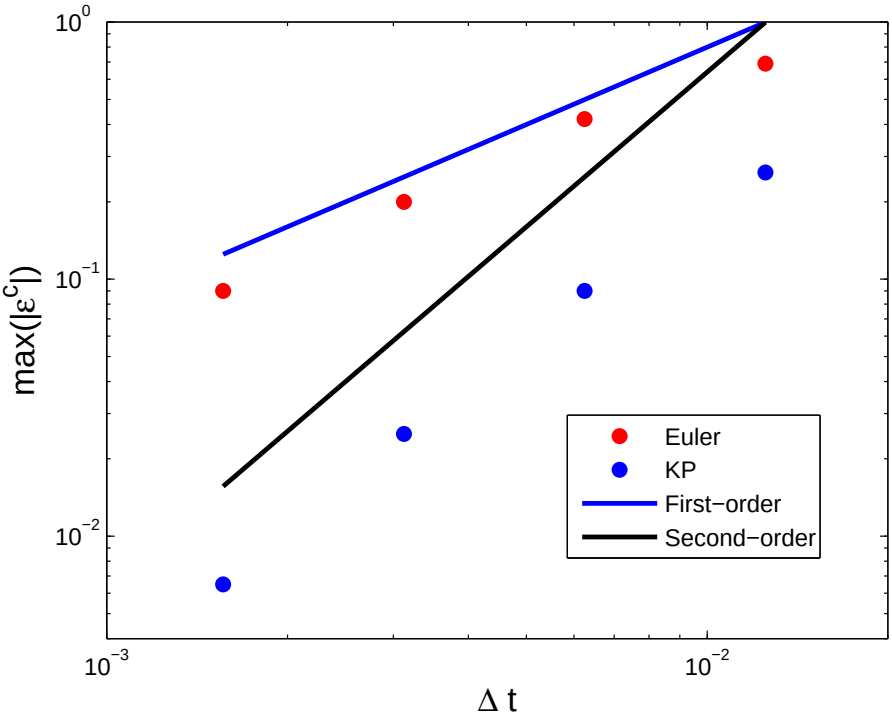

Figure 8: Results from the 2D PMC test case described in Appendix A. Maximal value of ϵ^C plotted against the time step.

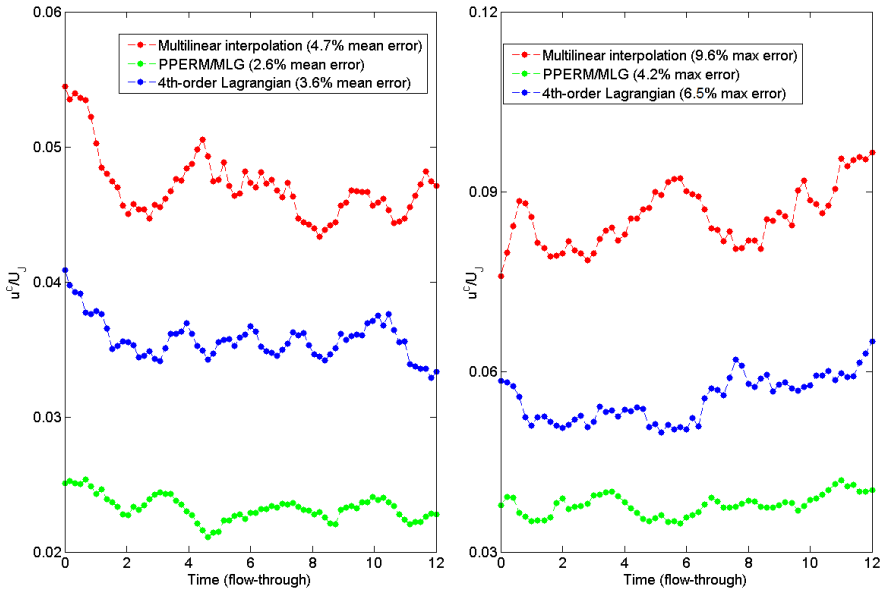

Figure 9: Time histories of the corrective velocity for an FC PPERM/MLG simulation with $n_p c = 30$, and an $F_3^{\alpha=3.0}(\epsilon^c)$ correction scheme with $\tau^c = 4\Delta t$. Left: L^1 measures of the corrective velocity. Right: L^∞ measures of the corrective velocity.

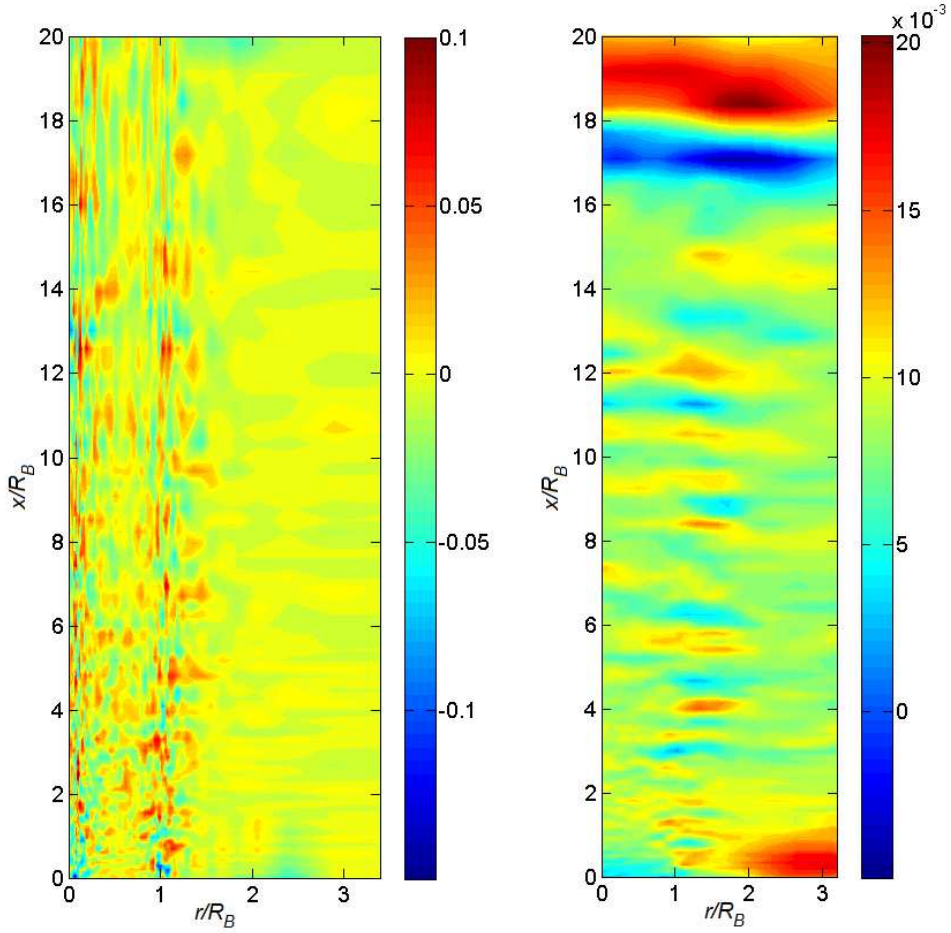

Figure 10: Contour plots, at the azimuthal location $\theta = 0$,of consistency error and corrective velocities for an FC PPERM/MLG simulation with $n_{pc} = 30$, and an $F_3^{\alpha=3.0}(\epsilon^c)$ correction scheme with $\tau^c = 4\Delta t$. Left: consistency error at the end of the simulation. Right: axial component of the normalized corrective velocity at the end of the simulation.

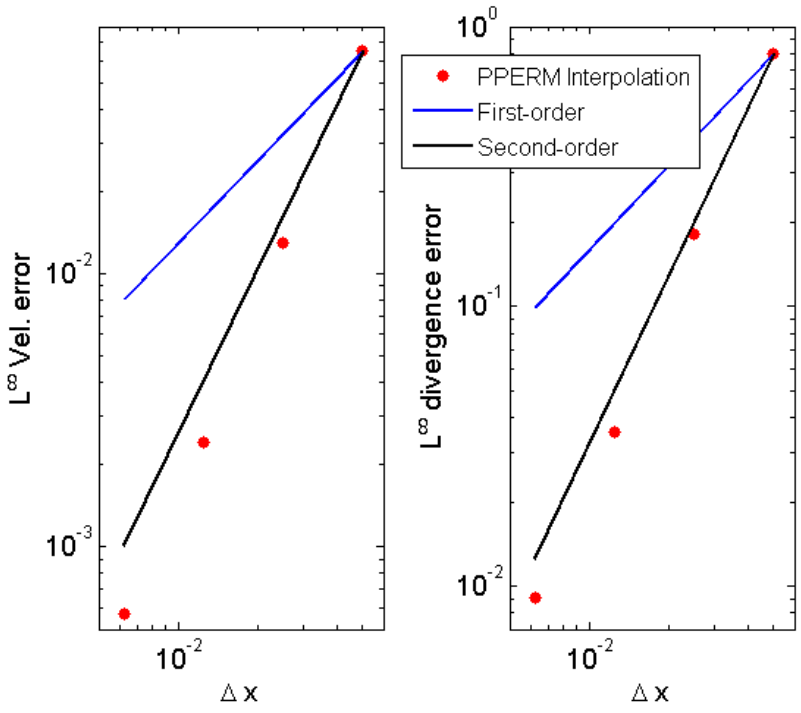

Figure 11: Convergence plots for the PPERM interpolation scheme when applied to the analytic test flow described at the end of Appendix B. Left: L^∞ errors for the interpolated velocity. Right: L^∞ errors for the divergence of the interpolated velocity.

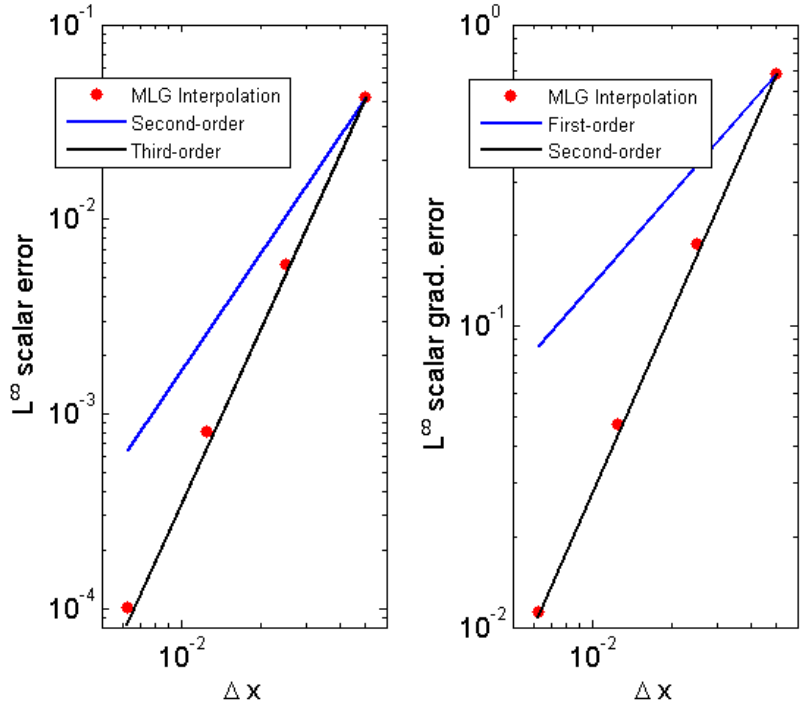

Figure 12: Convergence plots for the MLG interpolation scheme when applied to the analytic test flow described at the end of Appendix C. Left: L^∞ errors for the interpolated scalar. Right: L^∞ errors for the gradient of the interpolated scalar.

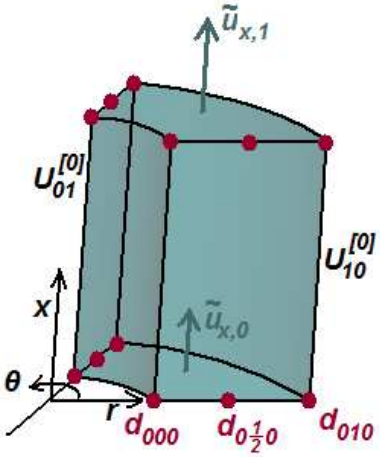

Figure 13: Illustration of the PPERM indexing notation. Single-subscript quantities, color-coded in teal, denote face-averaged information such as the face-average velocities $\tilde{u}_{x,0}$ and $\tilde{u}_{x,1}$ shown on the schematic. Double-subscript quantities, color-coded in black, denote PPERM edge coefficients, such as the 0^{th} order axial velocity coefficients $U_{10}^{[0]}$ and $U_{01}^{[0]}$ shown on the schematic. Triple-subscript quantities, color-coded in crimson, denote pointwise information at the 8 corners of the cell plus the 4 midpoints of the radial edges.

Large Eddy Simulation/Probability Density Function Simulations of Bluff Body Stabilized Flames

Pavel P. Popov

Stephen B. Pope

Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14853, USA

Dec 10, 2012

Abstract

This work presents large eddy simulation/probability density function (LES/PDF) simulation results for the Sandia/Sydney series of bluff-body stabilized CH_4/H_2 flames. Results are presented for the flames HM1, HM2 and HM3, using the 19-species ARM2 reduced chemical mechanism, and comparison is made with previous numerical simulations of the same flames. When compared to previous numerical studies of these bluff-body flames, the present simulation shows considerable improvement, particularly in the downstream regions of the flow. The simulations are shown to be sensitive to the treatment of heat transfer to the bluff-body face, with better agreement in the temperature profiles achieved with the addition of a Dirichlet temperature boundary condition.

1 Introduction

In the study of computational methods for turbulent reactive flows, the probability density function (PDF) chemistry modeling approach [12] is highly effective, due to the fact that there is no need for modeling of highly non-linear chemical source term [3]. In a large eddy simulation/probability density function (LES/PDF) algorithm [13], this advantage of the PDF chemistry model is coupled to the advantages of LES codes, which need no modeling for the large hydrodynamic scales which do not exhibit universal behavior [15]. As a result, modern LES/PDF codes are highly successful at simulating laboratory scale turbulent reactive flows [1,4,19, 18, 23].

In the present study, we apply a state of the art LES/PDF algorithm to the Sandia/Sydney series of CH_4/H_2 bluff body stabilized flames [8], in particular the flames HM1, HM2 and HM3. These flames feature a hydrodynamically

complex flow with a recirculation region attached to the bluff body face - a stabilization mechanism used in many technical applications - and local extinction for the cases HM2 and HM3. These features make the Sandia/Sydney bluff body flames both physically relevant and a natural application for an LES-based simulation, as opposed to a Reynolds-averaged Navier-Stokes-based solution.

The Sandia-Sydney bluff body series of flames, especially HM1, have previously been simulated by a variety of computational methods. Reynolds-averaged Navier-Stokes/Probability Density Function (RANS/PDF) solutions, using detailed chemistry (all the species in the chemical mechanism are tracked independently, subject to conservation of chemical elements), have been performed by Liu et al. [7] and Merci et al. [9]. A variety of large eddy simulation solutions exist, with chemistry modeling provided either via a steady-state flamelet model in the LES code, used by Kempf et al. [6], the direct quadrature method of moments used by Raman et al. [18], or via a particle probability density function (PDF) method, similar to the one used in the present study, either with detailed chemistry such as in the study of James et al. [5], or with a two-dimensional PDF sample space, consisting of mixture fraction and a reaction progress variable, in the work of Raman et al. [17,18].

Of the abovementioned works, [7] and [9] are the only ones which have performed simulations for the higher velocity flames HM2 and HM3 - the rest yield results for HM1 only. Previous researchers have found that the agreement with experimental data is best for the flame HM1, and deteriorates progressively for the faster flames HM2 and HM3, and also that the agreement is worse for locations which are far downstream in the axial direction.

In the present work, we perform LES/PDF simulations of the bluff body flames with reduced chemistry, using the ARM2 chemical mechanism, and compare our results with those of Liu et al. [7] and Merci et al. [9], as well as with the LES/PDF results of Raman et al. [17], which for the HM1 case yield best agreement with experiment among the studies listed above. As we shall see, the new computaional results show considerable improvement over these previous studies, especially in the downstream regions of the domain.

The rest of this paper is organized as follows: in section 2 we describe the equations governing the LES/PDF procedure used in the present study. Simulation details are provided in section 3, and the simulation results are presented and discussed in section 4. Finally, in section 5 we draw conclusions from the present set of simulation and their comparison to previous numerical results.

2 Equations Solved by the LES/HPDF Code

The LES/HPDF code used to simulate the bluff-body flame consists of two components: a finite volume (FV) LES code, based on a standalone-LES algorithm described by Pierce and Moin [10,17], and a Lagrangian particle PDF code, described by Wang and Pope [23]. The LES component of the LES/HPDF code solves the continuity and momentum equations

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho} \tilde{u}_j}{\partial x_j} = 0 \tag{1}$$

$$\frac{\partial \left(\overline{\rho} \tilde{u}_j\right)}{\partial t} + \frac{\partial (\overline{\rho} \tilde{u}_i \tilde{u}_j)}{\partial x_i} = -\frac{\partial \overline{p}}{\partial x_j} + 2\frac{\partial}{\partial x_i}\left(\overline{\rho}\left(\widetilde{\nu} + \widetilde{\nu}_T\right)\left(\tilde{S}_{ij} - \frac{1}{3}\tilde{S}_{kk}\delta_{ij}\right)\right), \tag{2}$$

where $\overline{p}$ and $\overline{\rho}$ are the LES resolved pressure and density, $\tilde{u}_j$, $\widetilde{\nu}$ and $\tilde{D}$ the Favre-averaged resolved velocity, molecular viscosity and molecular diffusivity, $\tilde{S}_{ij}$ is the resolved strain rate, and $\widetilde{\nu}_T$ and $\tilde{D}_T$ are the turbulent viscosity and diffusivity, respectively. The molecular viscosity and diffusivity are evaluated by the power law

$$\widetilde{\nu} = \nu_0 \left(\frac{\widetilde{T}}{300K}\right)^{1.69}, \ \frac{\widetilde{\nu}}{\widetilde{D}} = \sigma, \ \sigma = 0.82, \nu_0 = 1.42 \times 10^{-5} \left[\frac{m^2}{s}\right], \tag{3}$$

where the exponent in the power law is based on a curve fit to a CHEMKIN laminar flame calculation [23], and ν_0, σ are based on the properties of a stoichiometric fuel/air mixture at $300K$.

The turbulent viscosity and diffusivity are evaluated by the Dynamic Smagorinsky procedure, with Δ denoting the filter size (equal to the grid cell size) [2,10,11]. The LES component of the code also evaluates the scalar mixing frequency, Ω, defined as

$$\Omega = C_\phi \frac{\tilde{D}_T + 2\tilde{D}}{\Delta^2}, \tag{4}$$

which is used to model mixing in the PDF code, by the IEM procedure, with the mixing constant $C_\phi = 2.0$.

The PDF code advances the chemical compositions. In particular, we use ϕ_α to denote the composition vector, which in the present work consists of the specific moles of the 19 species in the ARM2 chemical mechanism, with the addition of enthalpy. Then, using ψ_α to denote points in the sample space of the composition vector ϕ_α, and using $f(\psi; \mathbf{x}, t)$ to denote the mass-weighted PDF of chemical compositions, conditional on the resolved velocity field [24,25], the modeled evolution equation for $f(\psi; \mathbf{x}, t)$ has the form

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x_i}\left(f\tilde{u}_i\right) = \frac{\partial}{\partial x_i}\left(\tilde{D}_T \frac{\partial f}{\partial x_i}\right) + \frac{\partial}{\partial \psi_\alpha}\left(f\Omega\left(\psi_\alpha - \tilde{\phi}_\alpha\right)\right)$$

$$-\frac{\partial}{\partial \psi_\alpha}\left(f\frac{1}{\bar{\rho}}\frac{\partial}{\partial x_i}\left(\bar{\rho}\tilde{D}\frac{\partial \tilde{\phi}_\alpha}{\partial x_i}\right)\right) + \frac{\partial}{\partial \psi_\alpha}\left(f S_\alpha(\psi)\right), \tag{5}$$

where the terms on the right hand side of eq.(5) are, in order, turbulent diffusion using the turbulent diffusivity hypothesis, turbulent mixing using the IEM mixing model [21], molecular diffusion, the reaction source term.

The evolution of eq.(5) is calculated via a Monte Carlo approximation [12], in order to avoid discretization in a high-dimensional space. The properties of an ensemble of Lagrangian particles are initialized throughout the computational domain and evolve by the following set of stochastic differential equations (SDEs):

$$dX_j^* = \left[\widetilde{u}_j + \frac{1}{\overline{\rho}} \frac{\partial \left(\overline{\rho} \widetilde{D}_T \right)}{\partial x_j} \right]^* dt + \left[2 \tilde{D}_T^* \right]^{1/2} dW_j^*, \tag{6}$$

$$d\phi_\alpha^* = -\Omega^* \left(\phi_\alpha^* - \widetilde{\phi}_\alpha^* \right) dt + \left[\frac{1}{\overline{\rho}} \frac{\partial}{\partial x_j} \left(\overline{\rho} \widetilde{D} \frac{\partial \widetilde{\phi}_\alpha}{\partial x_j} \right) \right]^* dt + S_\alpha \left(\boldsymbol{\phi}^* \right) dt. \tag{7}$$

where the term dW_j^* in eq.(6) denotes a Wiener increment, and the three terms on the right hand side of eq.(7) denote respectively turbulent mixing (in this case represented by the IEM model), molecular diffusion, and chemical reaction [22]. The superscript * in eqs.(6,7) is used to denote particle-based quantities. In the absence of numerical errors, the evolution of the mass-weighted PDF of the particle ensemble is identical to eq.(5), which is the PDF chemistry model which we aim to approximate numerically.

More details on the LES and PDF algorithms, and on their coupling, can be found in [10,23,16].

3 Simulation Details

The Sandia-Sydney Bluff-Body flames HM1, HM2 and HM3 consist of a fuel jet of diameter $3.6mm$ inside a bluff body of diameter $50mm$, itself located inside a square wind tunnel with sides of $150mm$. The fuel is $CH_4 : H_2$ in $1 : 1$ molar ratio, the coflow is air, and both fuel and coflow are at a temperature of $300K$. In all cases, the coflow bulk velocity is $40m/s$, and the fuel jet bulk velocity is $118m/s$ for the flame HM1, $178m/s$ for the flame HM2, and $212m/s$ for HM3; these fuel jet velocities correspond to 50%, 75% respectively 90% of the blowoff velocity [8].

The computational domain for the present simulation is $x \in [0, 20R_B]$, $r \in [0, 3.39R_B]$, where R_B is the bluff-body radius, and $r = 3.39R_B$ corresponds to a hydraulic diameter equal to that of the square wind tunnel. The grid size is $192 \times 128 \times 96$ in x, r, θ cylindrical coordinates - of the 128 cells in the radial direction, 85 discretize the bluff-body, 15 discretize the jet, and 28 discretize the coflow (the grid cell spacing in the radial direction increases considerably past $r = R_B$). Simulations are run for 30000 time steps of length $6 \times 10^{-6}s$ each, which corresponds to 14.5 flow-through times based on the coflow velocity; the number of particles per cell is set to 40.

The 19−species reduced chemical mechanism ARM2 [20] is used to model chemistry; this mechanism has previously been used successfully by Liu et al.

[7] for RANS/PDF simulations of the same series of flames. In Situ Adaptive Tabulation (ISAT) [14] is used for speedup of the chemical reaction calculations.

The velocity inlet boundary conditions for the fuel jet and the coflow are determined from an incompressible pipe flow simulation, with the pipe having either a circular or an annular cross section, for the jet and coflow, respectively. No-slip boundary conditions are enforced on the bluff-body surface and on the radial boundary of the cylindrical domain ($r = 3.39R_B$). We shall test two separate temperature boundary conditions for the bluff-body face. The first assumes that the bluff-body face is adiabatic, analogously to the simulations of Liu et al. [7], Merci et al. [9], and Raman et al. [17]; an optional Dirichlet temperature boundary condition has also been implemented by setting the temperature of all particles up to half a grid cell downstream axially from the bluff body to the mean bluff-body temperature measured from experimental data (respectively $953K$, $1007K$ and $1050K$ for the HM1, HM2 and HM3 flames). As we shall see in the next section, the inclusion of this boundary condition gives an improvement in the accuracy of temperature profiles close to the bluff-body over simulations without it, in which the bluff-body surface is adiabatic.

We note that the present simulation does not resolve the near-wall region, or use wall functions for heat transfer to the bluff-body face, and hence the Dirichlet temperature boundary condition described above is equivalent to taking the thermal resistance of the unresolved near wall-region to be zero. On the other hand, the standard adiabatic temperature boundary condition used in previous studies is equivalent to infinite thermal resistance in the near wall-region. These two boundary conditions, therefore, bound the actual thermal resistance of the near-wall region from above and below; as we shall see, in the present simulation the temperature profile yields better agreement with experiment when the Dirichlet boundary condition is used.

4 Results and Discussion

In this section, we present results for the bluff-body flames. Statistics are calculated over the second half of the simulation time interval, at which point the flow is statistically stationary; statistical averaging is performed in time and in the θ direction. Comparison is made with experimental data and, wherever the data are available, with previous computational results by Raman et al. [17], Liu et al. [7], and Merci et al. [9]. The simulation from Merci et al. which is used for comparison here is the one with the EMST mixing model and $C_\phi = 1.5$, whose Favre-averaged profiles give the best agreement with experimental data. For the flame HM1, we make a comparison between the temperatures obtained using the imposed temperature boundary condition discussed in the previous section, and those obtained without it (which imply that the bluff-body face is adiabatic). Unless otherwise specified, the results presented are from a simulation with the imposed temperature at the bluff-body face.

4.1 Results for Flame HM1

Figure 1 shows radial plots of Favre-averaged axial and radial velocity at three different axial locations, $x/D_B = 0.2, x/D_B = 0.6$ and $x/D_B = 1.4$. Here, the results are scaled by a factor of 108/118, in order to provide a valid comparison with the experimental velocity data provided by previous numerical studies, which are reported for the slightly slower flame HM1e, whose jet velocity is $108m/s$ instead of $118m/s$ for flame HM1. As we can see, all simulations give a good prediction of the mean axial velocity, with the exception of that of Merci et al., which considerably overpredicts the axial velocity close to the centerline for $x/D_B = 0.6$. The radial velocity plots show greater sensitivity to the simulation, and the agreement between simulations and experiment is worse here, especially at $x/D_B = 1.4$, where the two RANS/PDF calculations (Liu et al. and Merci et al.) underpredict the mean radial velocities by as much as $4m/s$ around $r = 10mm$. The LES calculations - that of the present study and that of Raman et al. - yield a better prediction of the mean radial velocity at that location, which is to be expected due to the advantages of LES over RANS in simulating more hydrodynamically complex flows, such as the present bluff-body flame. It should also be noted that the mean velocity profiles yielded by the two LES calculations are in close agreement with each other, which can be explained by the fact that both are based on the same incompressible LES solver and turbulence modeling described in Pierce and Moin [10], and that the grid resolution is similar in the two studies (Raman et al. use a $256 \times 128 \times 32$ grid).

Figure 2 shows radial plots of the root-mean-squares (RMS) of the axial and radial velocity fluctuations at the same axial locations: here, the RMS values are calculated based on the resolved velocity only. Again, the results are scaled by a factor of 108/118. Here the main difference between simulations and experiment is in the prediction of peak values of the rms velocity close to the centerline - at $x/D_B = 0.2$, the present simulation is in good agreement with the experimental rms axial velocity, and does not overpredicts the peak rms radial velocity as much as the simulation of Raman et al. At $x/D_B = 1.4$, all simulations underpredict the rms axial velocity for $r < 5mm$ and the rms radial velocity for $r < 10mm$.

Figures 3 and 4 show radial profiles of the Favre-averaged mixture fractions, and the RMS of its fluctuations respectively. Unlike the velocity RMS values, the mixture fraction RMS includes the contributions of both the resolved and residual mixture fraction fields. The present calculation is in generally in good agreement with experimental data at all axial locations - the same can be said of Raman et al.'s results. Merci et al.'s results are also in good agreement with experiment, except close to the centerline at $x/D_B = 0.6, 1.3$ where the peak mean mixture fraction is overpredicted. Finally, Liu et al. considerably underpredict the experimental data at the downstream locations $x/D_B = 1.8, 2.4$. We should also note that the present simulation overpredicts the mean mixture fraction at the upstream locations $x/D_B = 0.26, 0.6$, for $r > R_B$, where its mean mixture fraction drops off to zero more slowly than the experimental data and

the other calculations. This is most likely due to insufficient resolution in the outer shear layer, and as we shall see shortly, has an effect on the scalar profiles. While these discrepancies appear to be small, it should be appreciated that they appear around the stoichiometric mixture fraction, $\xi_{st} \approx 0.052$. Consequently, the impact on species may be more substantial, as is evident from the profiles of OH mass fraction presented below.

As we can see on fig. 4, similarly to the results for RMS of velocity fluctuations, the RMS of mixture fraction is hardest to predict correctly close to the centerline, for $r < 10mm$, where the present simulation overpredicts the experimental data in the middle axial locations $x/D_B = 0.9, 1.3$; our results also underpredict the peak RMS at $x/D_B = 2.4$, but are still in much better agreement than Liu et al.'s calculation, for which the location of peak RMS has shifted to the centerline.

Figure 5 shows radial profiles of Favre-averaged temperature. Here, we can see the effect of the temperature boundary condition on the bluff-body face: with the imposed experimentally measured temperature of $953K$, the temperature profiles at the upstream locations of $x/D_B = 0.26, 0.6$ are in much better agreement with experimental data, whereas without it the experimental data is overpredicred, by as much as $200K$ at $x/D_B = 0.6$. It can be seen that the present calculation (with the Dirichlet temperature boundary condition) compares favorably to other computational results. In particular, the simulation of Merci et al. considerably overpredicts (by roughly $250K$) the peak temperature values at $x/D_B = 0.6$, whereas that of Raman et al. underpredicts them (again, by about $250K$) at $x/D_B = 0.9, 1.3$. The calculations of Liu et al. are in good agreement with the experimental data up to $x/D_B = 1.3$, but for the last two axial locations the location of the peak in the radial profile is shifted towards the centerline.

We note that there is still considerable room for improvement in the present results. In particular, the temperature profiles at $x/D_B = 0.9, 1.3$ have large gradients near $r = 15mm$, unlike the experimental data and the other calculations. Also, at the upstream locations $x/D_B = 0.26, 0.6, 0.9$, the drop of the temperature profile from its peak to the coflow value is slower than that of experimental data and the other calculations, most likely due to the the more gradual drop in mean mixture fraction mentioned earlier. Therefore, it is expected that the agreement with experiment will be improved further by increased grid resolution at the outer shear layer.

Radial profiles of Favre-averaged CO mass fraction are shown on fig. 6. Here again, we can see the slower decrease in our simulation from the peak to the coflow value past $r = 25mm$. Also, at $x/D_B = 2.4$ the peak of CO mass fraction is shifted from $r = 10mm$ in the experimental data to $r = 5mm$, but the present results are still in better agreement at that location than those of Liu et al., which is the only other computational study to report mass fractions at that location. Upstream, at $x/D_B = 0.26$, both our simulation and that of Liu et al. are in good agreement with the peak CO mass fraction, whereas the calculation of Raman et al. underpredicts it by about 1% mass fraction; from here on, when discussing differences in mass fraction values, we shall use

the abbreviation MF to denote absolute values. For example, if we have an experimental value is 8%MF, and a computed value is 6%MF, we will either say that the computed value underpredicts experiment by 2%MF, or that it underpredicts experiment by 25%, the latter denoting relative differences.

Figure 7 shows radial profiles of Favre-averaged CO_2 mass fraction. The LES/HPDF calculation is generally in good agreement with the experiment, although the peak of the profile is consistently underpredicted by about 0.5%MF. Except at $x/D_B = 0.26$ and $x/D_B = 1.8$, where Raman et al.'s simulation better predicts the peak mass fraction, the current simulation gives an improvement over previous results.

Radial profiles of NO are shown on figure 8. Here we see some overprediction of the experimental results in both our simulation, but the agreement with experimental data is a condierable improvement over that of Liu et al. At the first four upstream locations, the present simulation reduces the overprediction of Liu et al.'s, approximately by half. At the two downstream locations, both computational profiles match experiment for $r > 15mm$, but for $r < 15mm$ the present simulation is considerably closer to the experimental profile than Liu et al.'s simulation. It is not surprising that this trend is similar to what was seen on the temperature profiles on figure 5, since NO formation is greater at higher temperatures.

Finally, fig. 9 gives radial profiles of Favre-averaged OH mass fraction. We see good agreement with experiment at most locations, with the exception of $x/D_B = 0.6$, where the present calculation gives a profile which is more spread out, and underpredicts the peak OH mass fraction by 0.6%MF. On the other hand, the agreement with experimental data for $x/D_B = 0.26$, $r < 17mm$ is much better than that of the other two simulations for which data are available, and downstream, at $x/D_B = 1.3, 1.8, 2.4$, the agreement of our simulation and that of Raman et al. is much better than that of Liu et al.

Figures 10 and 11 show, for the axial locations $x/D_B = 0.6, 0.9$ and $x/D_B = 1.3, 1.8$ respectively, scatter plots of temperature vs. mixture fraction, for the mixture fraction interval $\xi \in [0, 0.2]$ which includes the stoichiometric value of $\xi = 0.052$. For the reader's reference, a laminar flamelet profile is also shown, calculated via CHEMKIN's OPPDIF module, using the ARM2 chemical mechanism and a strain rate of $30s^{-1}$. Note that there appears to be local extinction at the location $x/D_B = 0.6$. This is in fact inert mixing in the outer shear layer, as pointed out by Liu et al.: for the LES/HPDF calculation, all the data points in the low temperature band lie in the region $r/R_B > 0.94$. The agreement of the LES/HPDF conditional means to those of the experiment is good, with the exception that the present simulation does not capture the sharp decrease in the conditional mean between the stoichiometric point and $\xi = 0.07$. We should note that Liu et al. [7], and Merci et al. [9] also provide temperature scatter plots at some of these locations, which cannot be reproduced for comparison here.

Contour plots of resolved axial velocity and temperature (with and without the temperature Dirichlet boundary condition) at the end of the simulation are provided on figures 12, 13, and 14 respectively. Due to the hydrodynami-

cally complex nature of the bluff-body flow, with vortex shedding at both shear layers, we can see that the instantaneous velocity and temperature fields are considerably more complex than what is suggested by the Favre-averaged radial plots. Also, a comparison of figures 13 and 14 illustrates the significant effect of the bluff-body temperature boundary condition on the upstream temperature field.

4.2 Results for Flame HM2

In this subsection, we present results for flame HM2, and make a comparison with experimental data and the calculations of Liu et al. and Merci et al.

A comparison of mean and RMS mixture fractions can be seen on figures 15 and 16, respectively. Similarly to the HM1 case, the LES/HPDF mean mixture fraction is generally in good agreement with experiment, again with the exception of a slower decrease to the coflow value past $r = 25mm$ and some underprediction of the experimental data for $x/D_B = 1.8, 2.4$, $r < 10mm$. In contrast, the calculations of Liu et al. considerably underpredict the mean mixture fraction at $x/D_B = 1.8, 2.4$, and overpredict it close to the bluff-body face, ar $x/D_B = 0.26$, $5mm < r < 25mm$. The calculations of Merci et al. give better agreement with experiment than those of Liu et al., but not as good as the LES/HPDF simulation: the peak values at the centerline are overpredicted, and the profile is less spread out in the radial direction for $x/D_B = 1.8$.

On the RMS mixture fraction plots (fig. 16), we again see that the largest differences between simulations and experiment are in the peak RMS values close to the centerline. These are well captured by the LES/HPDF calculation and that of Merci et al. (though in the latter, the peak at $x/D_B = 1.8$ is closer to the centerline than in the experiment), and considerably overpredicted by the simulation of Liu et al.

Radial profiles of Favre-averaged temperature are shown on fig. 17. On the two upstream plots of $x/D_B = 0.26$ and 0.6 we see that the LES/HPDF profile is in good agreement with experimental data, whereas the two RANS/PDF simulations of Liu et al. and Merci et al. underpredict temperature by as much as $200K$. At $x/D_B = 0.9$ and 1.3, the agreement between LES/HPDF and experiment is somewhat poorer, with the location of the peak temperature closer to the centerline. Downstream, at $x/D_B = 1.8$ and 2.4 there is again good agreement with experiment, particularly when compared to the simulation of Liu et al., which considerably overpredicts the experimental profile. We note again the presence in the LES/HPDF simulations of the artifacts seen in the HM1 results, namely a slower decrease of the profile past $r = 25mm$ for the upstream locations, and a high gradient near $r = 10mm$ for the $x/D_B = 0.9$ profile.

Figure 18 shows radial profiles of mean CO mass fraction. Here, we see that both simulations - LES/HPDF and that of Liu et al. - are in good agreement with experimental data, except at $x/D_B = 1.8$, where the results of Liu et al. considerably overpredict the CO mass fraction close to the centerline.

Radial profiles of the mean CO_2 mass fraction are shown on fig. 19. Similarly

to the HM1 case, the LES/HPDF similation is in good agreement with the experiment - the peak mass fraction is underpredicted by about 1%MF at the first three locations, and overpredicted by approximately 0.5%MF at $x/D_B = 1.3$ and $x/D_B = 1.8$. This trend (underprediction upstream, overprediction downstream) is more pronounced in the simulation of Liu et al., but the agreement of that calculation with experiment is better here than in the HM1 case.

Mean NO mass fraction profiles are shown on fig. 20. The most significant difference from the results for the HM1 case is that both simulations accurately predict the experimental results at $x/D_B = 0.26$. Similarly to the HM1 case, at the $x/D_B = 0.6$ and $x/D_B = 0.9$ locations, the LES/HPDF simulation somewhat overpredicts experimental results, by up to 20% of the experimental value, and this discrepancy is doubled in the Liu et al. simulation. Downstream, LES/HPDF provides a considerable improvement over the results of Liu et al., which are significantly higher than the experiment close to the centerline.

Finally, OH mass fractions are shown on fig. 21. Here, we see that for the upstream locations $x/D_B = 0.26, 0.6$ and 0.9, the LES/HPDF profiles correctly capture the peak OH mass fractions, but are more spread out in the radial direction, and again, the decrease past $r = 25mm$ is slower. For those locations, Liu et al.'s results give better agreement with experiment. The opposite is true of the downstream locations $x/D_B = 1.8$ and 2.4 where LES/HPDF simulations correctly capture the location of peak OH mass fraction, whereas the calculations of Liu et al. predict a peak closer to the centerline.

Scatter plots of temperature vs. mixture fraction are shown on figures 22 and 23, for the axial locations $x/D_B = 0.6, 0.9$ and $x/D_B = 1.3, 1.8$ respectively. At $x/D_B = 0.6$, the temperature variation near the stoichiometric point is significantly lower in the LES/HPDF simulation than in the experiment, and the experimentally measured maximal temperatures (up to $2400K$) are considerably underpredicted - the LES/HPDF maximum is $2250K$. Apart from the location $x/D_B = 0.6$, the agreement of the LES/HPDF simulation with experimental data is good, both in the conditional means and in the fact that the present simulation features considerable local extinction at $x/D_B = 1.8$, similarly to the experimental data, though above $\xi = 0.1$ the lower bound of the experimental scatter plot is overpredicted by up to $150K$. Similarly to the HM1 case, the sharp decrease in the conditional mean from the stoichiometric point to $\xi = 0.07$ is not replicated in the LES/HPDF simulation.

4.3 Results for Flame HM3

In this subsection, we present results for flame HM3, and make a comparison with experimental data and the calculations of Liu et al. and Merci et al.

A comparison of mean and RMS mixture fractions can be seen on figures 24 and 25, respectively. Similarly to the HM1 and HM2 cases, the LES/HPDF mean mixture fraction is generally in good agreement with experiment, again with the exception of a slower decrease to the coflow value past $r = 25mm$ and some underprediction of experiment for $x/D_B = 1.8$ and 2.4, $r < 10mm$. Similarly to the HM2 case, Liu et al. considerably underpredict the mean mixture

fraction at $x/D_B = 1.8, 2.4$, and overpredict it close to the bluff-body face, ar $x/D_B = 0.26$, $5mm < r < 25mm$. The calculations of Merci et al. give better agreement with experiment than those of Liu et al., although the peak values at the centerline are overpredicted, and the profile is less spread out in the radial direction for $x/D_B = 1.8$.

On the RMS mixture fraction plots (fig. 25), we again see that the largest differences between simulations and experiment are in the peak RMS values close to the centerline. These are well captured by the LES/HPDF calculation and that of Merci et al. (again, the peak at $x/D_B = 1.8$ is closer to the centerline than experiment), and considerably overpredicted by the simulation of Liu et al.

Radial profiles of Favre-averaged temperature are shown on fig. 26. On the upstream plot of $x/D_B = 0.26$ we see that the LES/HPDF profile underpredicts the experimental data, but is in better agreement than the calculations of Liu et al. At $x/D_B = 0.6$ both LES/HPDF and the calculations of Liu et al. are in good agreement with experiment, whereas Merci et al. underpredict the temperature profile by as much as $400K$. In the two intermediate axial locations LES/HPDF and Liu et al. are again in good agreement with experiment. The profile of Liu et al. becomes more inaccurate downstream, and overpredicts experiment by as much as $600K$ at $x/D_B = 2.4$. Unlike in the HM1 and HM2 cases, the LES/HPDF profile, too, loses some of its accuracy downstream, overpredicting the peak temperature value by $100K$ at $x/D_B = 1.8$ and by $150K$ at $x/D_B = 2.4$. Still, the agreement with experiment is much better than that of the RANS/PDF simulations.

Figure 27 shows radial profiles of mean CO mass fraction. Here, we see that both simulations - LES/HPDF and that of Liu et al. - are in good agreement with experimental data downstream, except at $x/D_B = 1.8$, where the results of Liu et al. considerably overpredict the CO mass fraction close to the centerline. At the two upstream locations, the LES/HPDF simulation overpredicts experimental data by as much as 1%MF, but is still considerably more accurate than the simulation of Liu et al., which overpredict experiment by as much as 5%MF.

Mean CO_2 mass fraction profiles are shown on figure 28. At $x/D_B = 1.3$ and downstream from that location, the results are similar to the HM1 and HM2 cases - LES/HPDF slightly overpredicts the experimental profile, by up to 10% of the experimental values, whereas Liu et al. are in good agreement with experiment for for $r > 10mm$ but considerably overpredict experimental values close to the centerline. At the three upstream locations, particularly at $x/D_B = 0.26$, the agreement, for both the present simulation and Liu et al., is poorer than in the HM1 and HM2 cases: in particular, the LES/HPDF simulation underpredicts experiment by up to 1%MF at $x/D_B = 0.26$. Still, the present results for the most part predict experimental data better than those of Liu et al.

Figure 29 shows radial profiles of mean NO mass fractions. Again, the agreement of the LES/HPDF simulation with experimental data is poorer than that in the HM1 and HM2 cases, with the computational values overpredicting ex-

periment by a factor of more than 2 downstream. Still, there is an improvement over the previous RANS calculation, especially at $x/D_B = 0.6$ and $x/D_B = 0.9$ where, unlike in the HM1 and HM2 cases, Liu et al.'s calculation overestimates experimental results by more than a factor of 2.

Radial profiles of mean OH mass fractions are shown on fig. 30. Again, the agreement of the LES/HPDF results with experiment is not as good as in the HM1 and HM2 cases, particularly at $x/D_B = 1.8, 2.4$ where the LES/HPDF profile overpredicts the peak OH mass fraction, but is considerably better than that of Liu et al.'s profiles, especially at $x/D_B = 0.26$ where the latter profile considerably underpredicts the OH mass fraction for $r < 20mm$.

Scatter plots of temperature vs. mixture fraction are shown on figures 31 and 32, for the axial locations $x/D_B = 0.6, 0.9$ and $x/D_B = 1.3, 1.8$ respectively. Similarly to the HM2 case, the maxima on the temperature scatter plot for $x/D_B = 0.6$ are underpredicted in the LES/HPDF simulation, although the discrepancy at the stoichiometric point is not quite as pronounced. We can observe significant local extinction at $x/D_B = 0.9, 1.3$ and 1.8. The agreement in the conditional means is good at $x/D_B = 0.6$ and $x/D_B = 0.9$, but deteriorates somewhat at $x/D_B = 1.8$, where the LES/HPDF conditional means around $\xi = 0.07$ overpredict experiment by up to $100K$, and the local extinction is somewhat underpredicted. We note that experimental conditional means are not shown for $x/D_B = 1.3$ because they are not present in the experimental data set.

4.4 Further discussion

From the results presented in the above two subsections, we see that the LES/HPDF simulations of the flames HM1, HM2 and HM3 are in very good agreement with experiment, with the exception of the upstream locations past $r = 25mm$, where the LES/HPDF profiles decrease to the coflow values more slowly than the experimental data, most probably due to insufficient resolution in the outer shear layer. The LES/HPDF results are also a considerable improvement over other simulations of the same flames, especially at the downstream locations of $x/D_B = 1.8$ and 2.4.

As can be seen on the scatter plots of temperature vs. mixture fraction, the present simulation is able to capture the local extinction in the HM2 and HM3 cases, although the amount of local extinction is somewhat underpredicted in the downstream regions of the HM3 case. Nevertheless, the local extinction of flame HM3 is captured in the LES/HPDF calculation much better than in previous simulations: in particular, both the simulations of Liu et al. and the EMST simulations of Merci et al. considerably underpredict the amount of local extinction in the flames HM2 and HM3. For the flame HM2, another simulation reported in Merci et al., using a modified Curl's mixing model [9], yields scatter plots in better agreement with experimental data (in particular, the local extinction at $x/D_B = 1.8$ for the HM2 case is well-predicted), but gives more inaccurate mean field results, and cannot produce a stable burning solution for the HM3 case.

The consequences of the improved prediction of local extinction can be observed on figure 33, which plots the progression from flame HM1 to HM3, as given by experimental data, and as predicted by the present simulation and that of Liu et al. We can see that the experimental data indicates a sharp decrease in temperature at $x/D_B = 0.9$ from HM2 to HM3, as well as a progressive decrease in temperature from HM1 to HM2 to HM3 at the downstream locations $x/D_B = 1.3$, $x/D_B = 1.8$ and $x/D_B = 2.4$. These trends are approximated well by the LES/HPDF calculation, but not so much by that of Liu et al.

5 Conclusions

The Sandia-Sydney bluff-body flames have been simulated with a new LES/PDF algorithm, called LES/HPDF, developed by Cornell University's Turbulence and Combustion Group. Using the skeletal ARM2 chemical mechanism, the computational results are in very good agreement with experiment. A notable exception to this is in the outer shear layer close to the bluff-body, where the LES/HPDF-calculated scalars converge to the coflow value more slowly than experimental data and other computational results; additionally the NO mass fractions are generally overpredicted, especially downstream in the HM3 flame, but the agreement with experiment is still considerably better than in previous work. Overall, the present calculations are a substantial improvement on previous computational studies of the same flame, with better prediction of the local extinction in flames HM2 and HM3. It is seen that the simulation is sensitive to the treatment of heat transfer to the bluff body face, and better agreement of the temperature profiles with experiment is seen with the implementation of a Dirichlet temperature boundary condition which enforces the experimentally-observed mean temperature on the bluff-body face.

Acknowledgements

This work was supported by Office of Energy Research, Office of Basic Energy Sciences, Chemical Sciences, Geosciences and Biosciences Division of the US Department of Energy (DOE) under Grant DE-FG02-90ER-14128.

References

[1] P. Colucci, F. Jaberi, P. Givi, and S. B. Pope, *Filtered Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, Phys. Fluids **10** (1998), pp.499-515

[2] M. Germano, U. Piomelli, P. Moin, and W.H. Cabot *A dynamic subgrid-scale eddy viscosity model*, Phys. Fluids, **3**, pp. 1760-1765, 1991

[3] D. C. Haworth, *Progress in Probability Density Function Methods for Turbulent Reacting Flows*, Prog. Energy Comb. Sci., **36**, 2, (2010), pp. 168-259

[4] F. Jaberi, P. Colucci, S. James, P. Givi, and S. B. Pope, *Filtered Mass Density Function for Large Eddy Simulation of Turbulent Reacting Flows*, J. Fluid Mech. **401** (1999), pp.85-121

[5] S. James, J. Zhu, M. Anand, and B. Sekar, *Large Eddy Simulations of Bluff-Body Stabilized Turbulent Flames and Gas Turbine Combustors*, DoD High Performance Computing Modernization Program Users Group Conference, Pittsburgh, PA, June 18-21

[6] A. Kempf, R. P. Lindstedt, and J. Janicka, *Large-Eddy Simulation of a Bluff-Body Stabilized Nonpremixed Flame*, Comb. Flame, **143**, (2006), pp. 170-189

[7] K. Liu, S. B. Pope, and D. A. Caughey, *Calculations of Bluff-Body Stabilized Flames Using a Joint Probability Density Function Model with Detailed Chemistry*, Comb. Flame, **141**, (2005), pp. 89-117

[8] A. R. Masri, and R. W. Bilger, *Turbulent Diffusion Flames of Hydrocarbon Fuels Stabilized on a Bluff Body*, Proc. Comb. Inst., **20**, (1985)

[9] B. Merci, D. Roekaerts, B. Naud, and S. B. Pope, *Comparative Study of Micromixing Models in Transported Scalar PDF Simulations of Turbulent Nonpremixed Bluff Body Flames*, Comb. Flame, **146**, (2006), pp. 109-130

[10] C. D. Pierce, and P. Moin, *Progress-Variable Approach for Large-Eddy Simulation of Turbulent Combustion*, J. Fluid Mech.**504**, (2004), pp. 73-97

[11] P. Moin, K. Squires, W. Cabot, and S. Lee, *A Dynamic Subgrid-Scale Model for Compressible Turbulence and Scalar Transport*, Phys. Fluids**3**, (1991), pp. 2746-2757

[12] S. B. Pope, *PDF Methods for Turbulent Reactive Flows*, Prog. Energy Combust. Sci., **11**, pp. 119-192 (1985)

[13] S. B. Pope *Computations of Turbulent Combustion: Progress and Challenges*, Proc. Comb. Inst., **23**, (1991), pp. 591-612

[14] S. B. Pope, *Computationally Efficient Implementation of Combustion Chemistry Using In Situ Adaptive Tabulation*, Comb. Theory Model., **1**, (1997), pp. 41-63

[15] S. B. Pope, *Turbulent Flows*, Cambridge University Press, Cambridge, 2000

[16] P. P. Popov, H. Wang, and S. B. Pope, *Specific Volume Coupling and Convergence Properties in Hybrid Particle/Finite Volume Algorithms for Turbulent Reactive Flows*, J. Comp. Phys. (in review)

[17] V. Raman, H. Pitsch, and R. O. Fox, *Hybrid Large-Eddy Simulation/Lagrangian Filtered-Density-Function Approach for Simulating Turbulent Combustion*, Comb. Flame, **143**, (2005), pp. 56-78

[18] V. Raman, H. Pitsch, and R. O. Fox, *Eulerian Transported Probability Density Function Sub-Filter Model for Large-Eddy Simulations of Turbulent Combustion*, Comb. Theory Model., **10**, 3, (2006), pp. 439-458

[19] V. Raman, and H. Pitsch, *A Consistent LES/Filtered Density Function Formulation for the Simulation of Turbulent Flames with Detailed Chemistry*, Proc. Comb. Inst., **31**, (2007), pp. 1711-1719

[20] Q. Tang, Ph.D. Thesis, Cornell University, 2003

[21] J. Villermaux, and J.C. Devillon *Reprèsentation de la redistribution des domaines de sègrègation dans un fluide par un modèle d'interaction phènomènologique*, 2nd Int. Symp. Chem. React. Engng, Amsterdam, 1972

[22] S. Viswanathan, H. Wang, and S. B. Pope, *Numerical Implementation of Mixing and Molecular Transport in LES/PDF Studies of Turbulent Reacting Flows*, J. Comput. Phys. **230**, (2011), pp. 6916-6957

[23] H. Wang, and S. B. Pope *Large Eddy Simulation/Probability Density Function Modeling of a Turbulent $CH_4/H_2/N_2$ Jet Flame*, Proc. Comb. Inst., **33**, (2011), pp. 1319-1330

[24] S. B. Pope *Self-Conditioned Fields for Large-Eddy Simulations of Turbulent Flows*, J. Fluid Mech., **652**, (2010), pp. 139-169

[25] R. O. Fox *Computational Models for Turbulent Reactive Flows*, Cambridge University Press, Cambridge, 2003

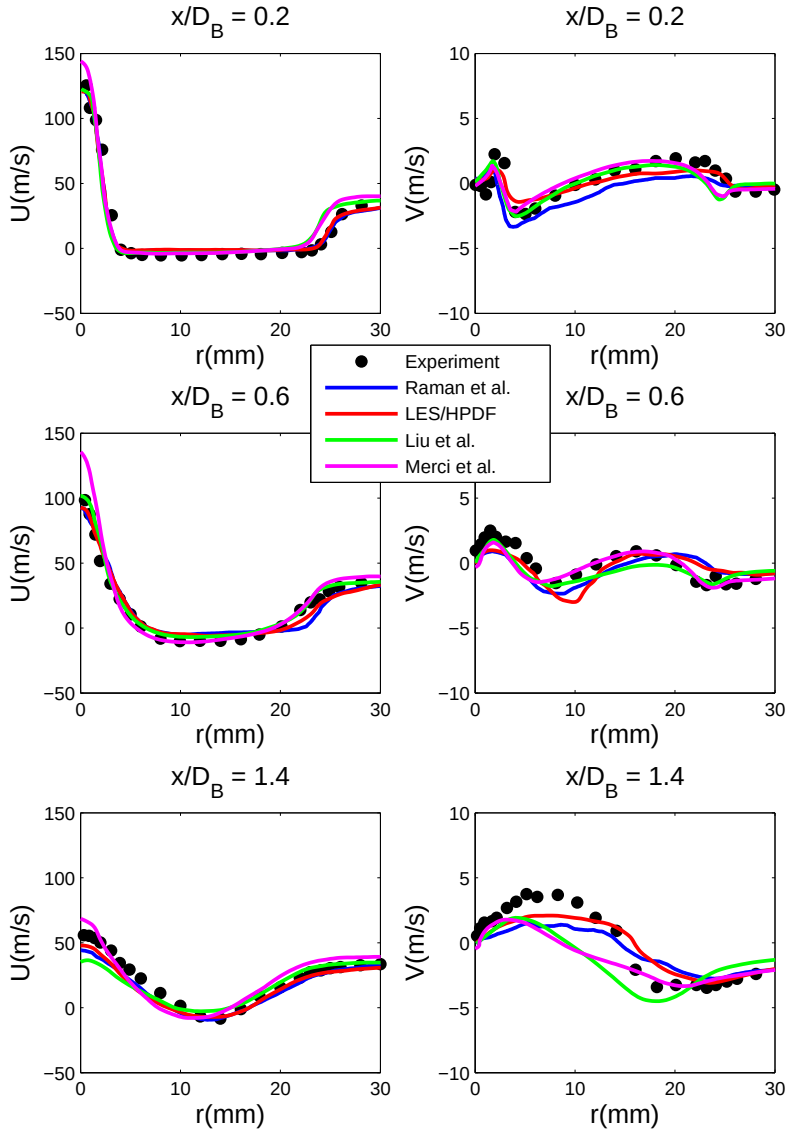

Figure 1: Mean velocity plots for the HM1e simulation. Left: radial plots of the Favre-averaged axial velocity at three different axial locations: $x = 0.2D_B, x = 0.6D_B$, and $x = 1.4D_B$. Right: plots of the Favre-averaged radial velocity at the same locations. As noted in the text, the LES/HPDF profiles are scaled by a factor of 108/118, in order to account for the velocity difference between flames HM1 and HM1e.

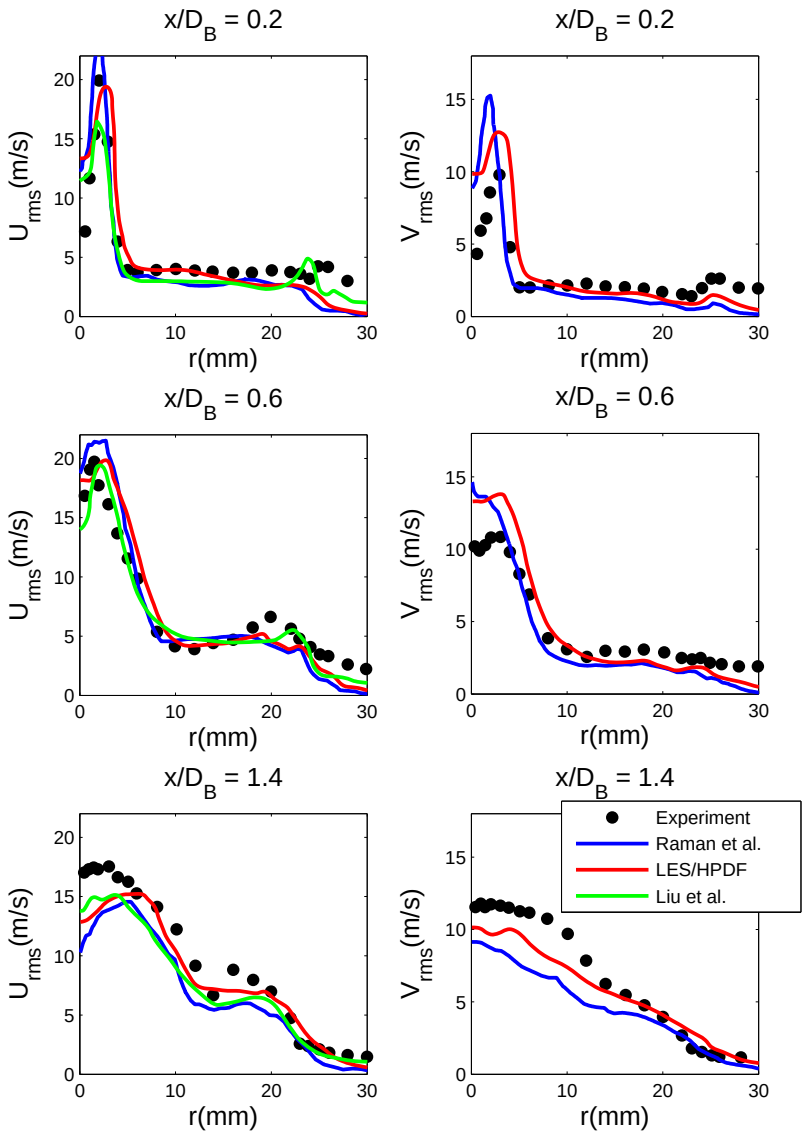

Figure 2: RMS velocity plots for the HM1e simulation. Left: radial plots of the root-mean-square of axial velocity fluctuations at three different axial locations: $x = 0.2D_B, x = 0.6D_B$, and $x = 1.4D_B$. Right: plots of the root-mean-square of axial velocity fluctuations at the same locations. As noted in the text, the LES/HPDF profiles are scaled by a factor of 108/118, in order to account for the velocity difference between flames HM1 and HM1e.

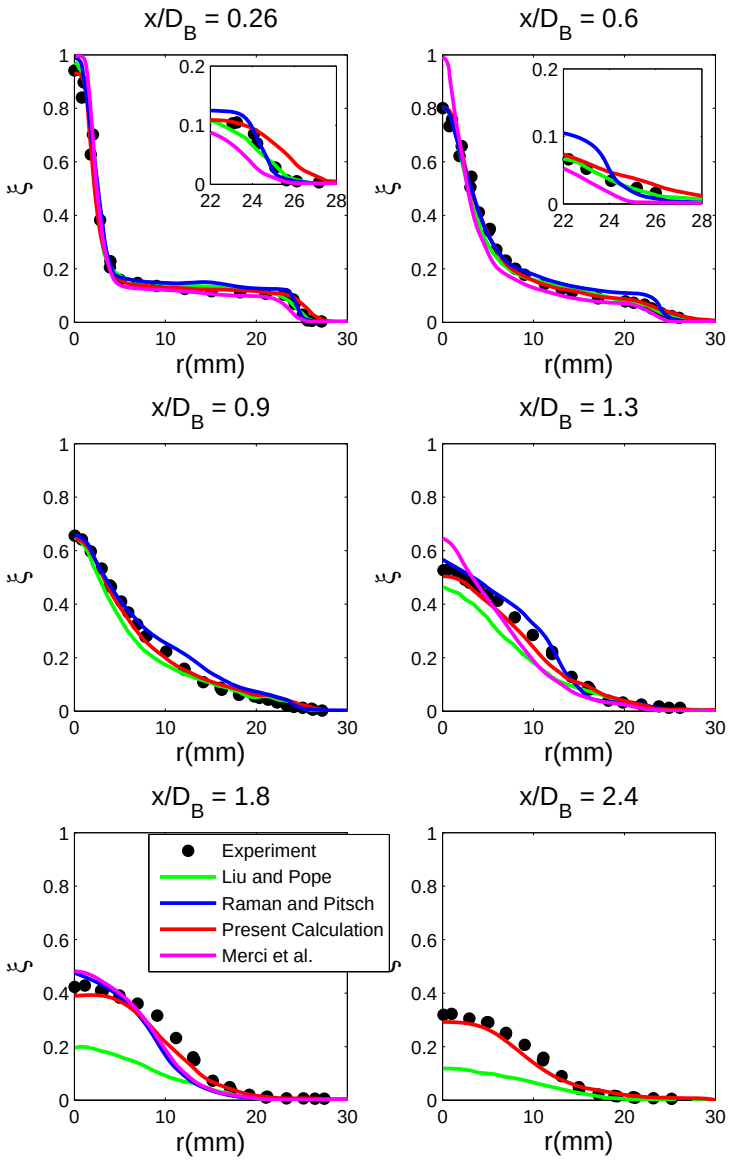

Figure 3: Radial plots of the Favre-averaged mixture fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

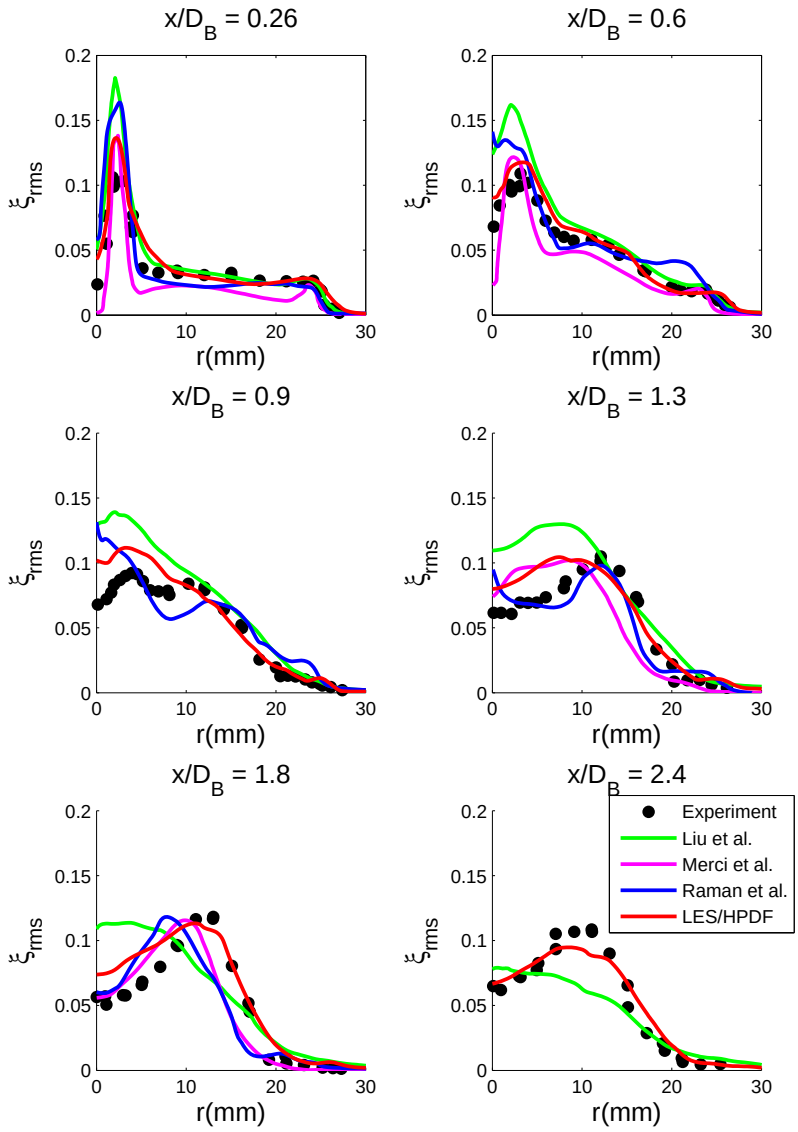

Figure 4: Radial plots of the root-mean-square of mixture fraction fluctuations at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

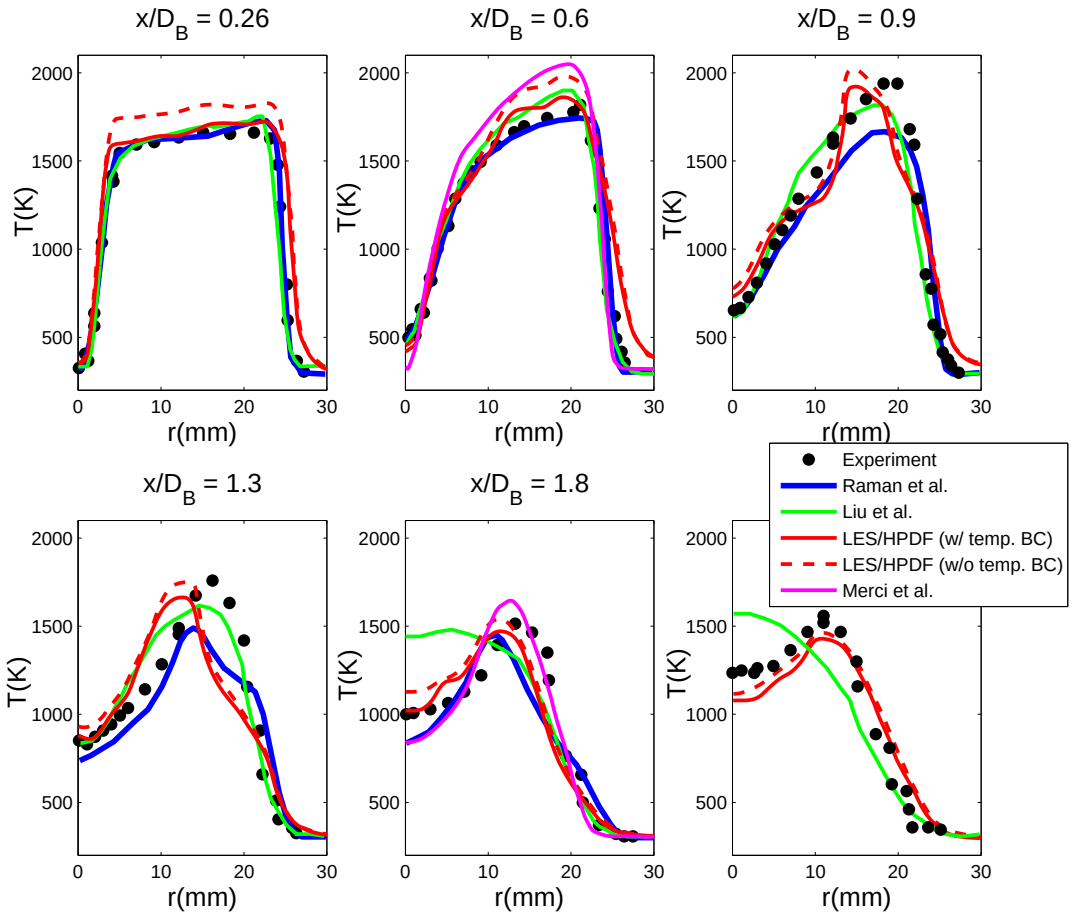

Figure 5: Radial plots of the Favre-averaged temperature at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation, with and without the temperature boundary condition at the bluff-body face plate.

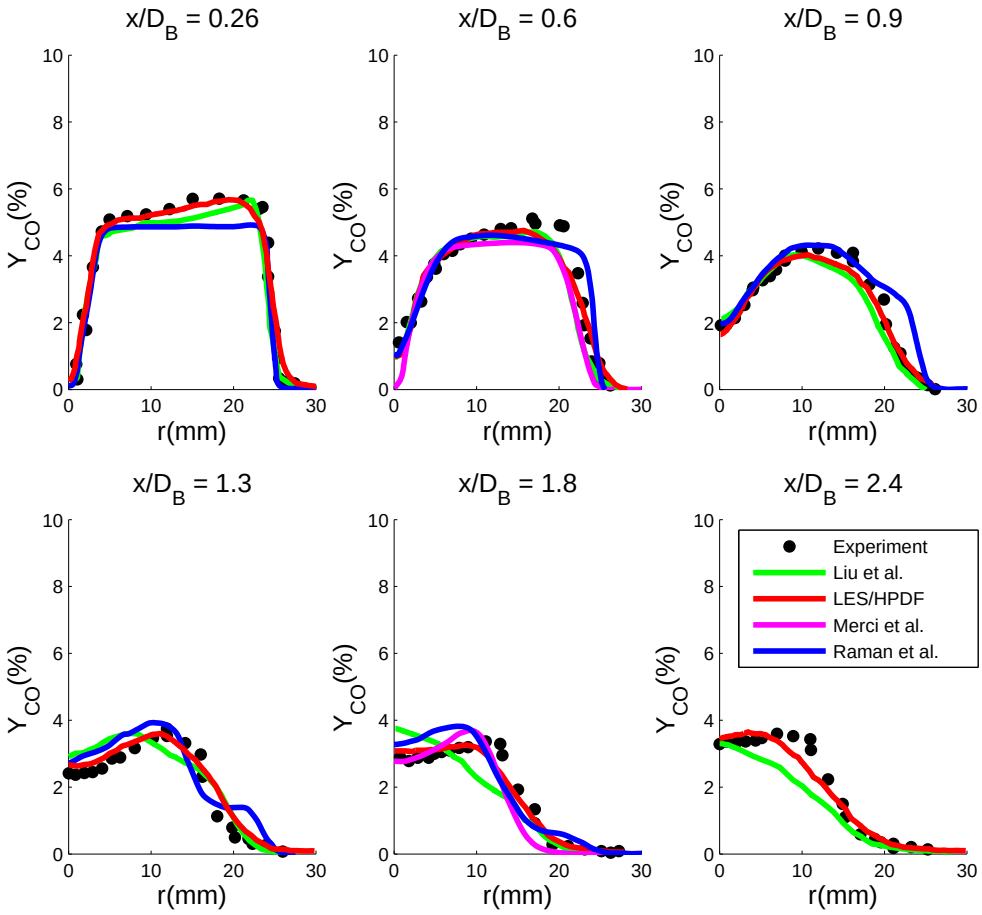

Figure 6: Radial plots of the Favre-averaged CO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

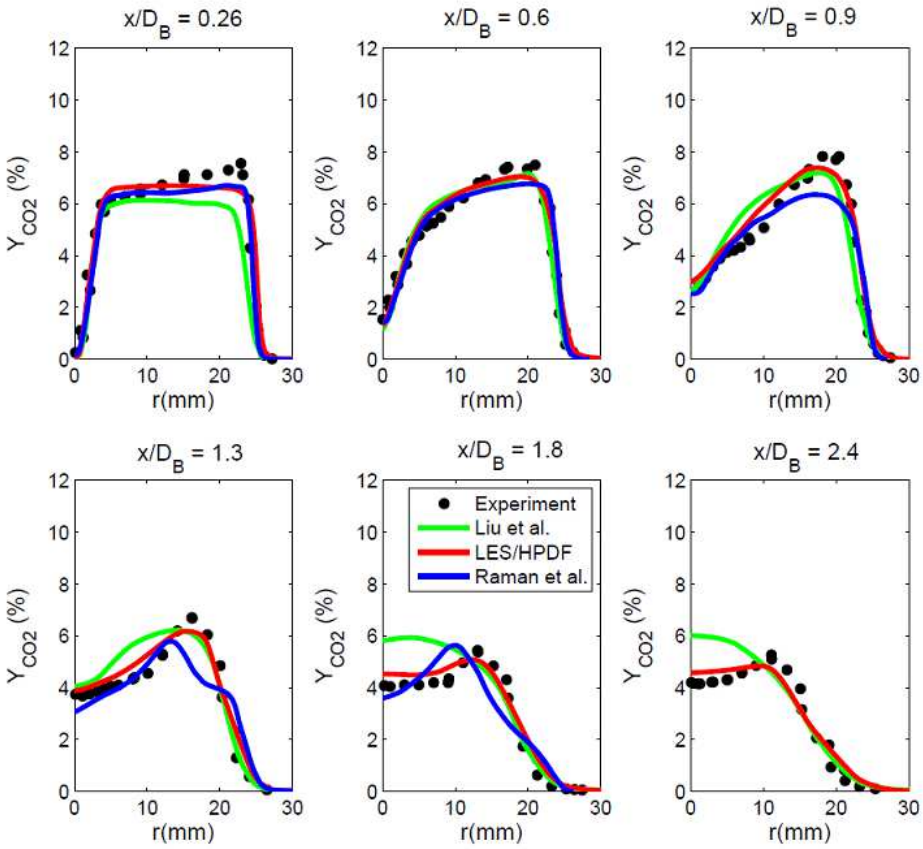

Figure 7: Radial plots of the Favre-averaged CO2 mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

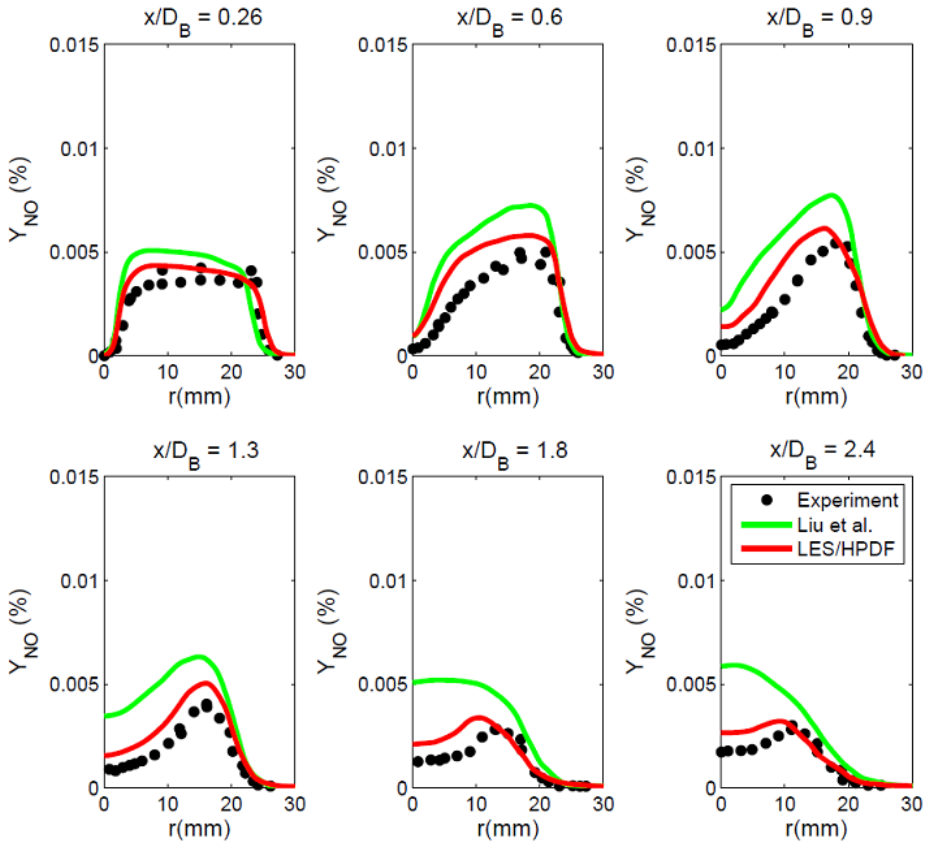

Figure 8: Radial plots of the Favre-averaged NO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

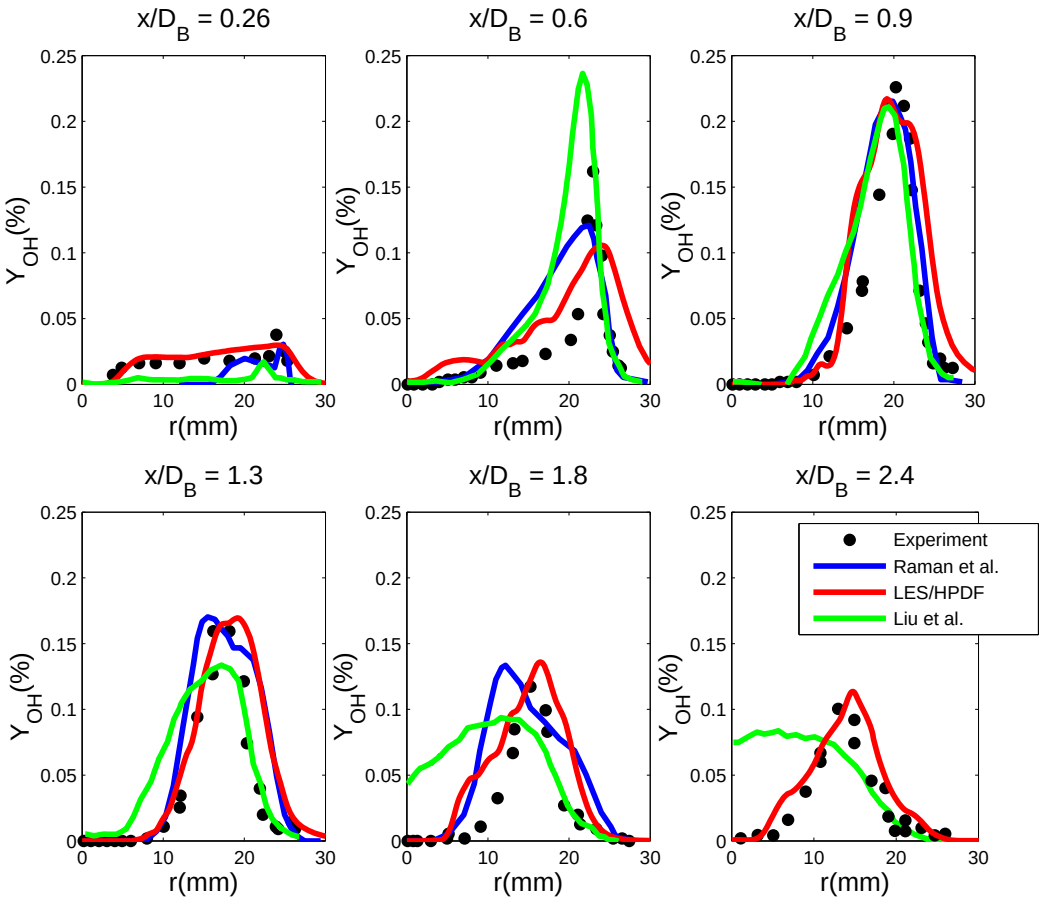

Figure 9: Radial plots of the Favre-averaged OH mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM1 simulation.

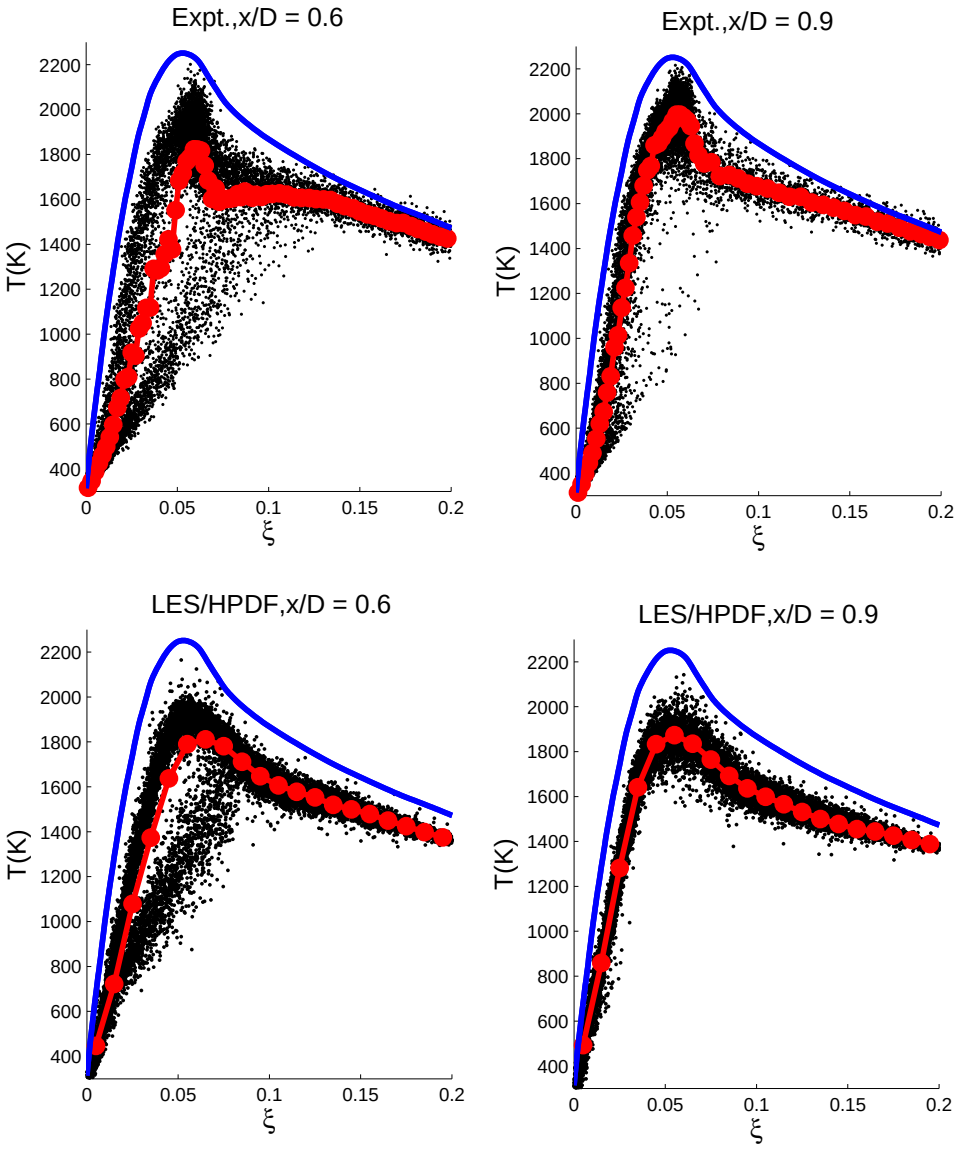

Figure 10: HM1 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 0.6D_B$ and $x = 0.9D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

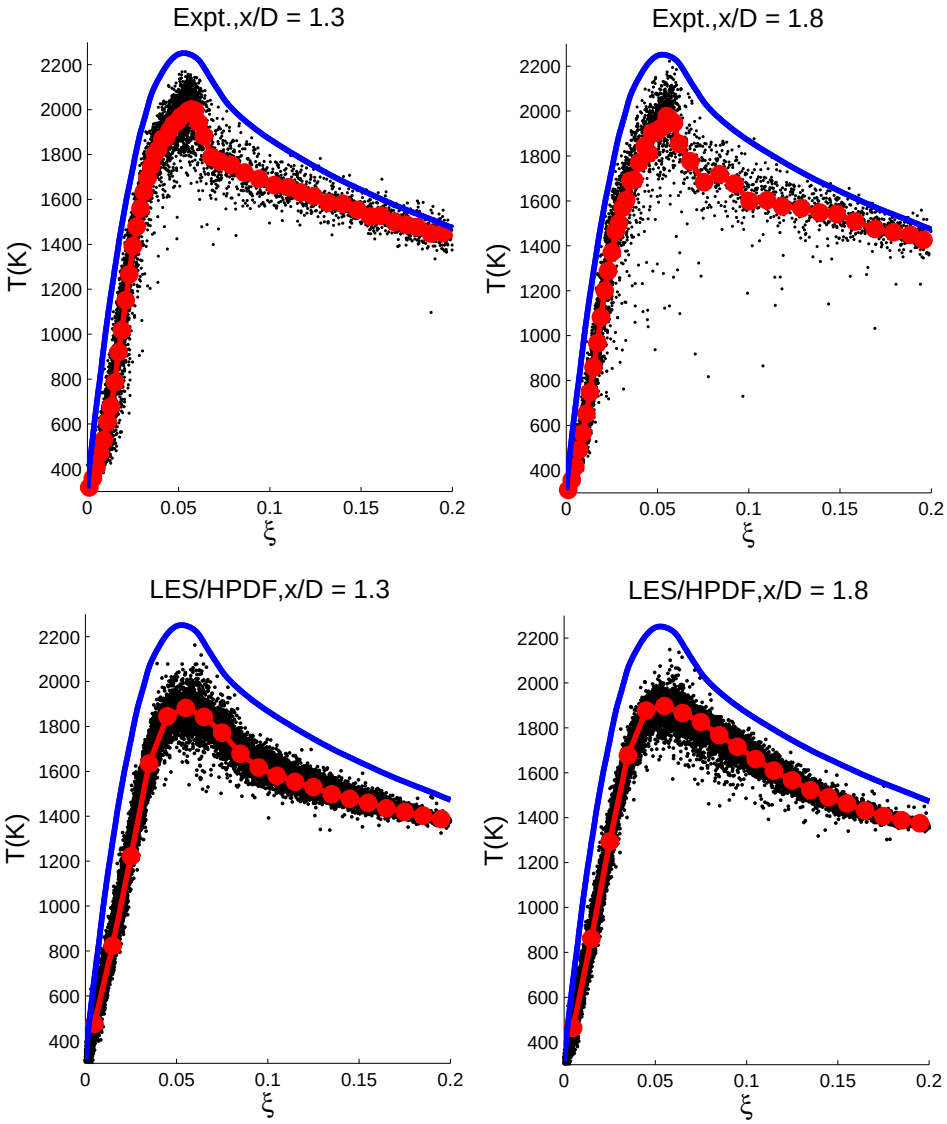

Figure 11: HM1 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 1.3D_B$ and $x = 1.8D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

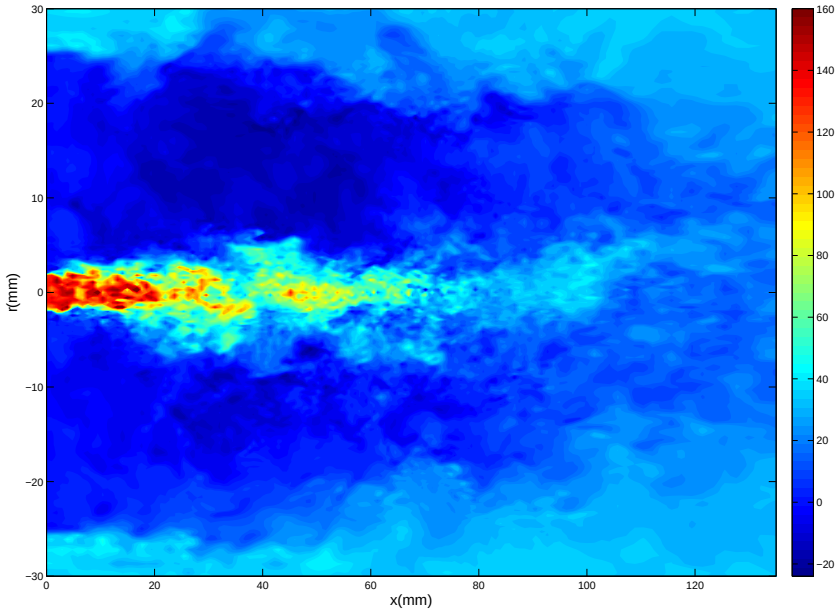

Figure 12: Contour plot of the resolved axial velocity (in m/s) at the end of the HM1 simulation.

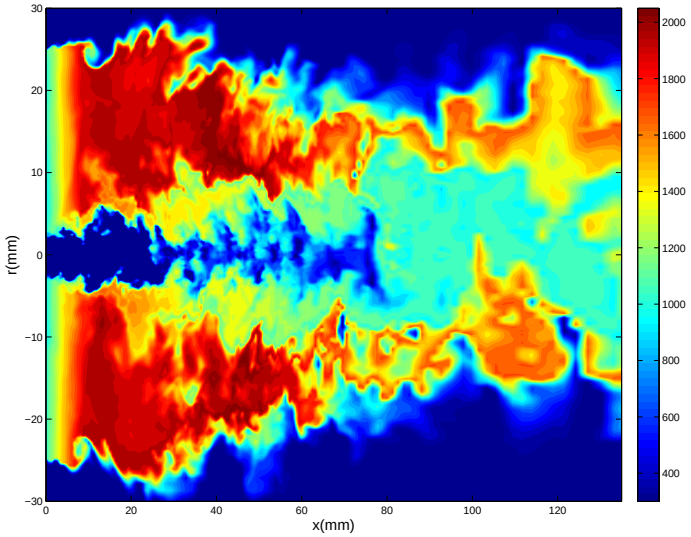

Figure 13: Contour plot of the resolved temperature (in K) at the end of the HM1 simulation, with a specified temperature at the bluff-body face.

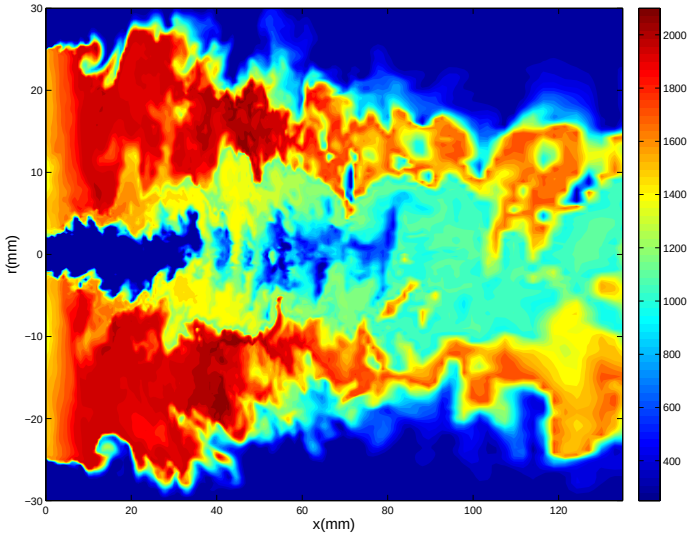

Figure 14: Contour plot of the resolved temperature (in K) at the end of the HM1 simulation, without the temperature boundary condition (bluff-body face is adiabatic).

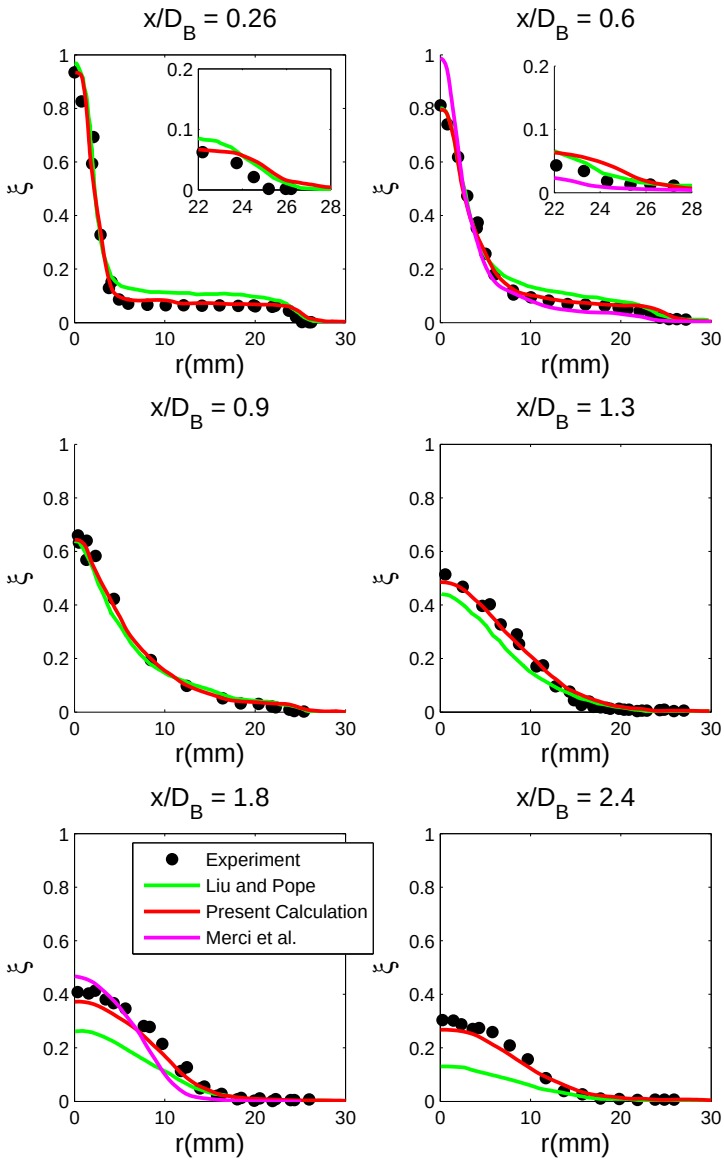

Figure 15: Radial plots of the Favre-averaged mixture fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

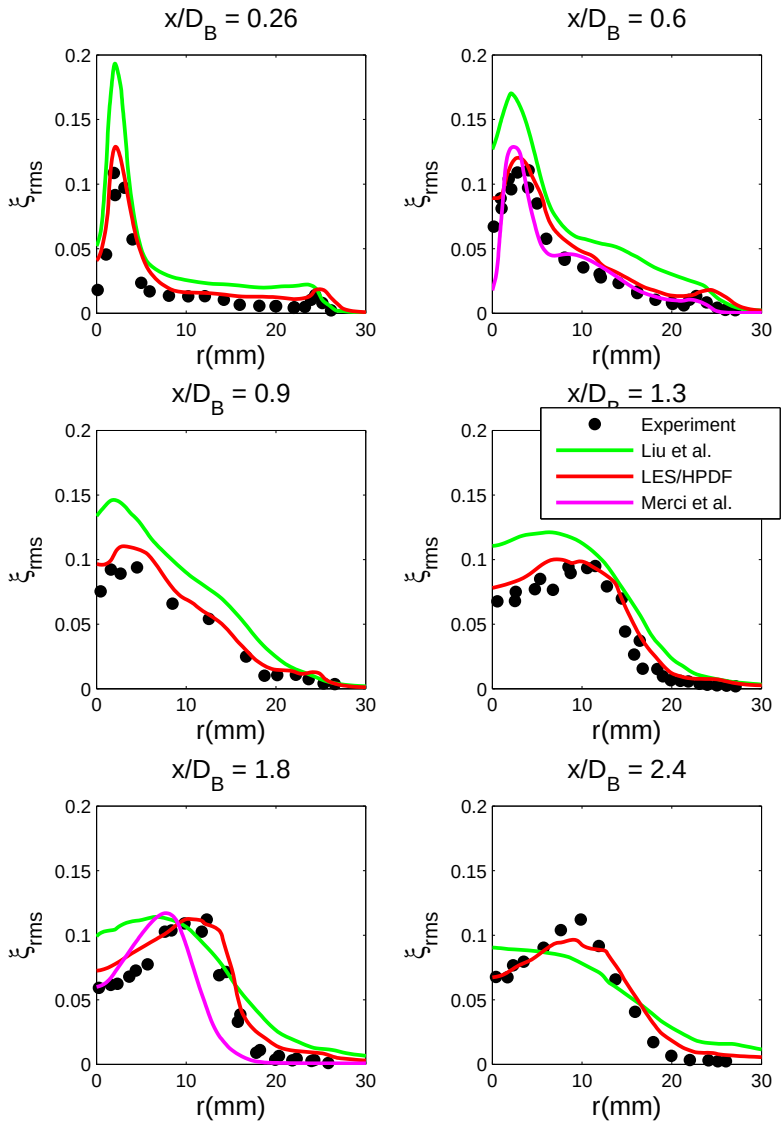

Figure 16: Radial plots of the root-mean-square of mixture fraction fluctuations at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

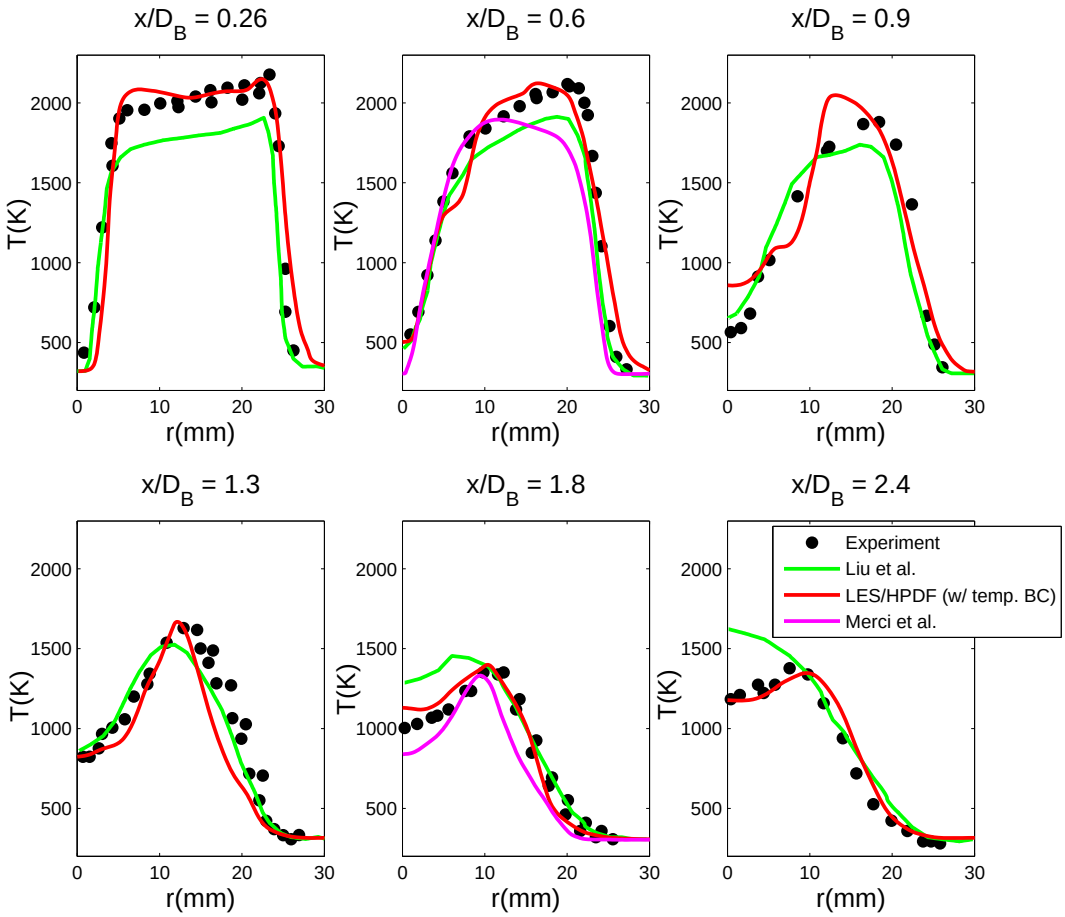

Figure 17: Radial plots of the Favre-averaged temperature at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

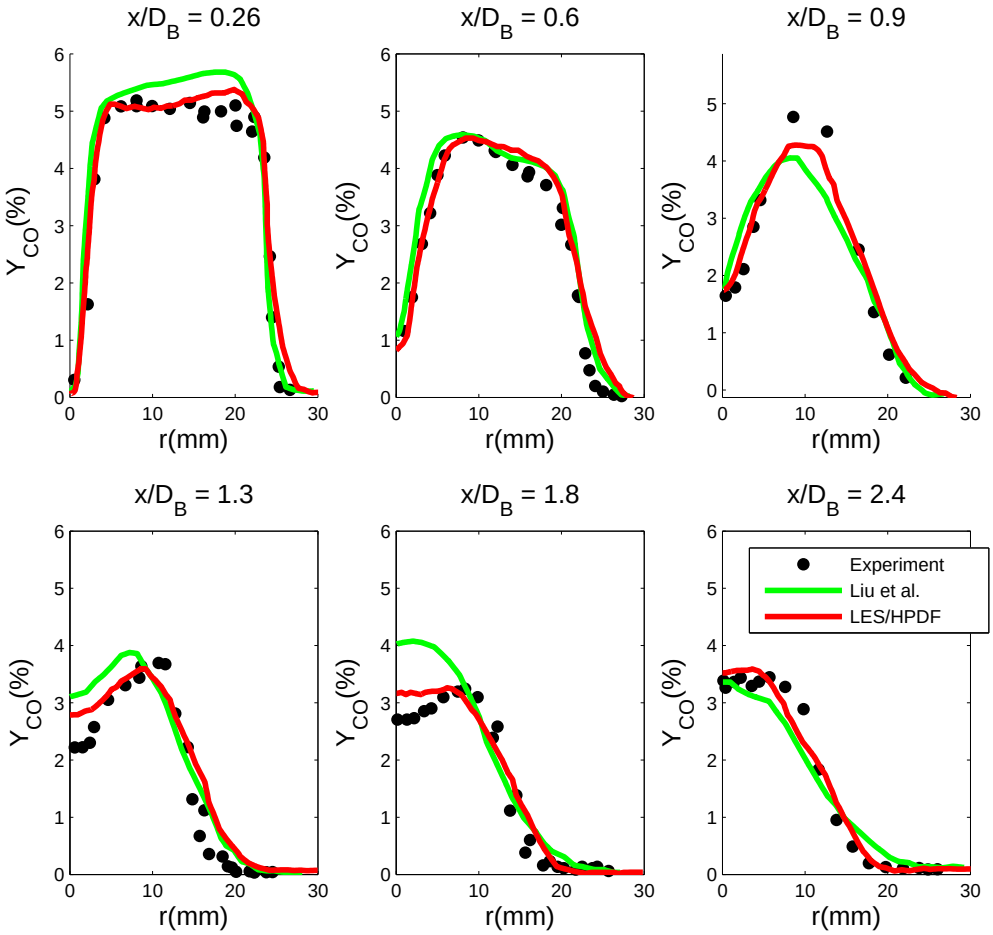

Figure 18: Radial plots of the Favre-averaged CO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

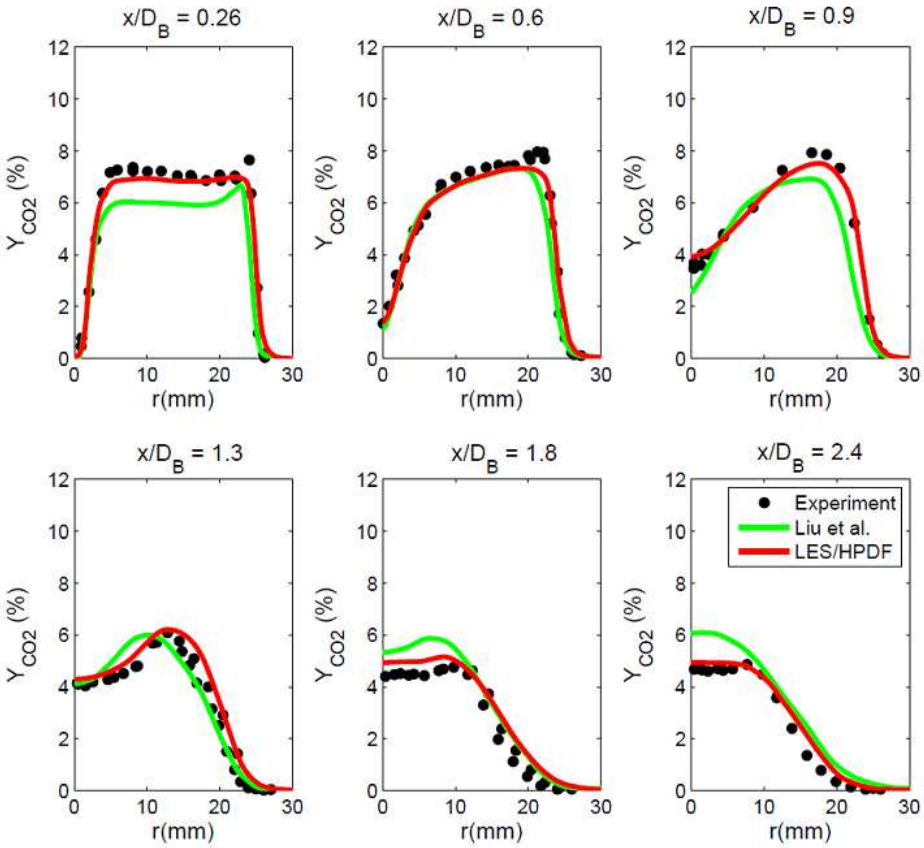

Figure 19: Radial plots of the Favre-averaged CO2 mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

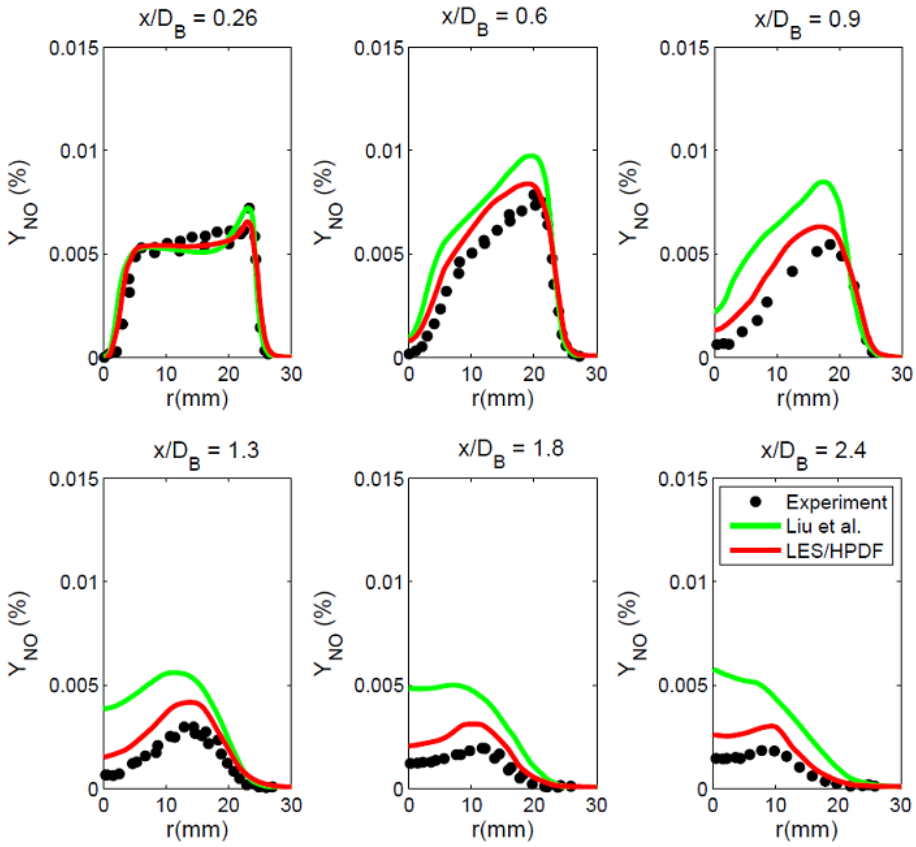

Figure 20: Radial plots of the Favre-averaged NO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

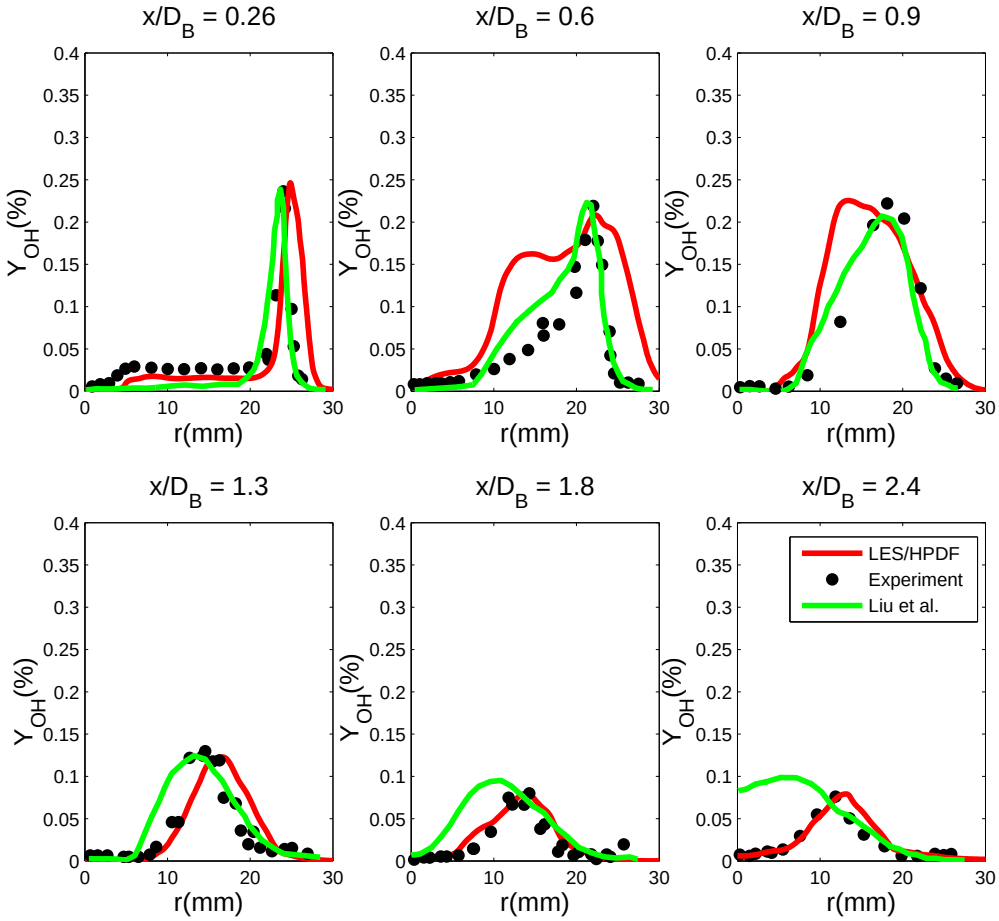

Figure 21: Radial plots of the Favre-averaged OH mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM2 simulation.

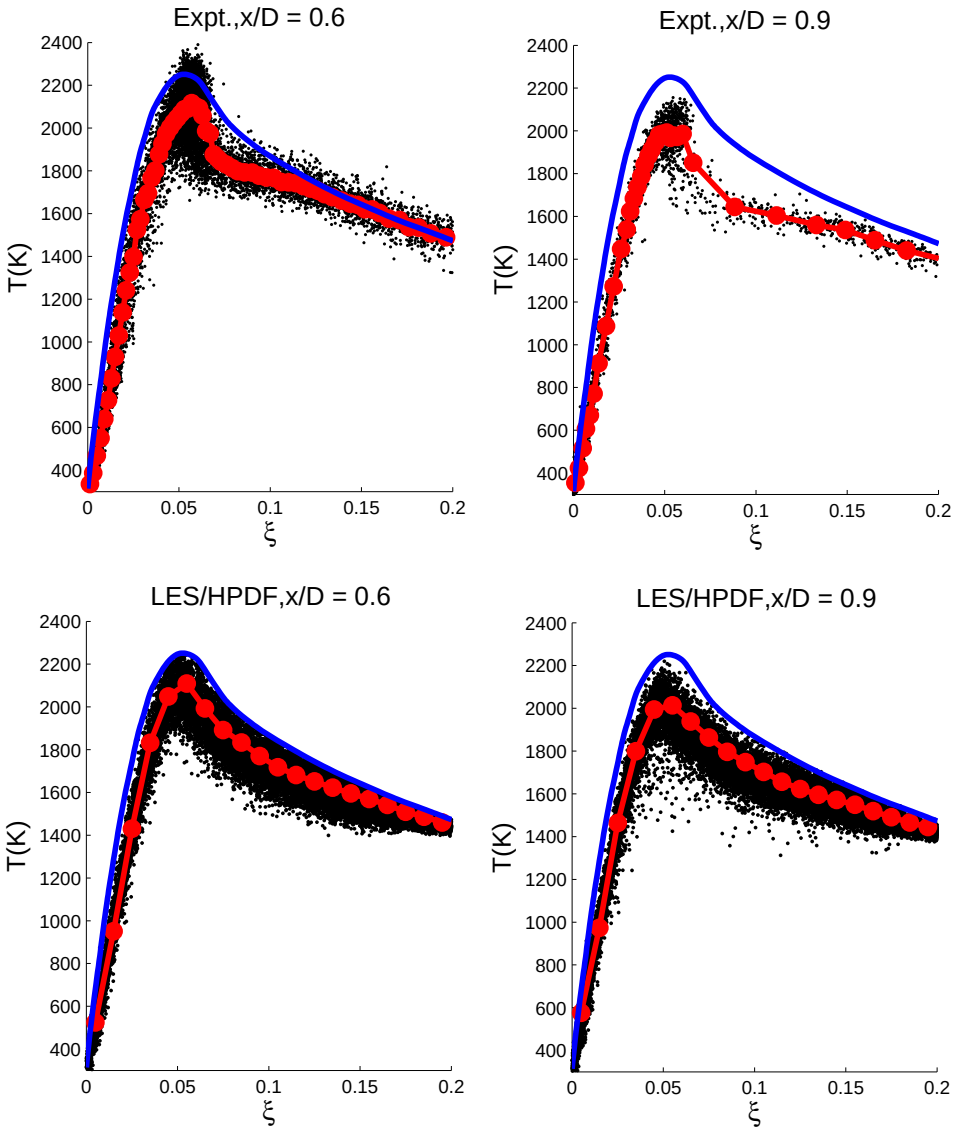

Figure 22: HM2 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 0.6D_B$ and $x = 0.9D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

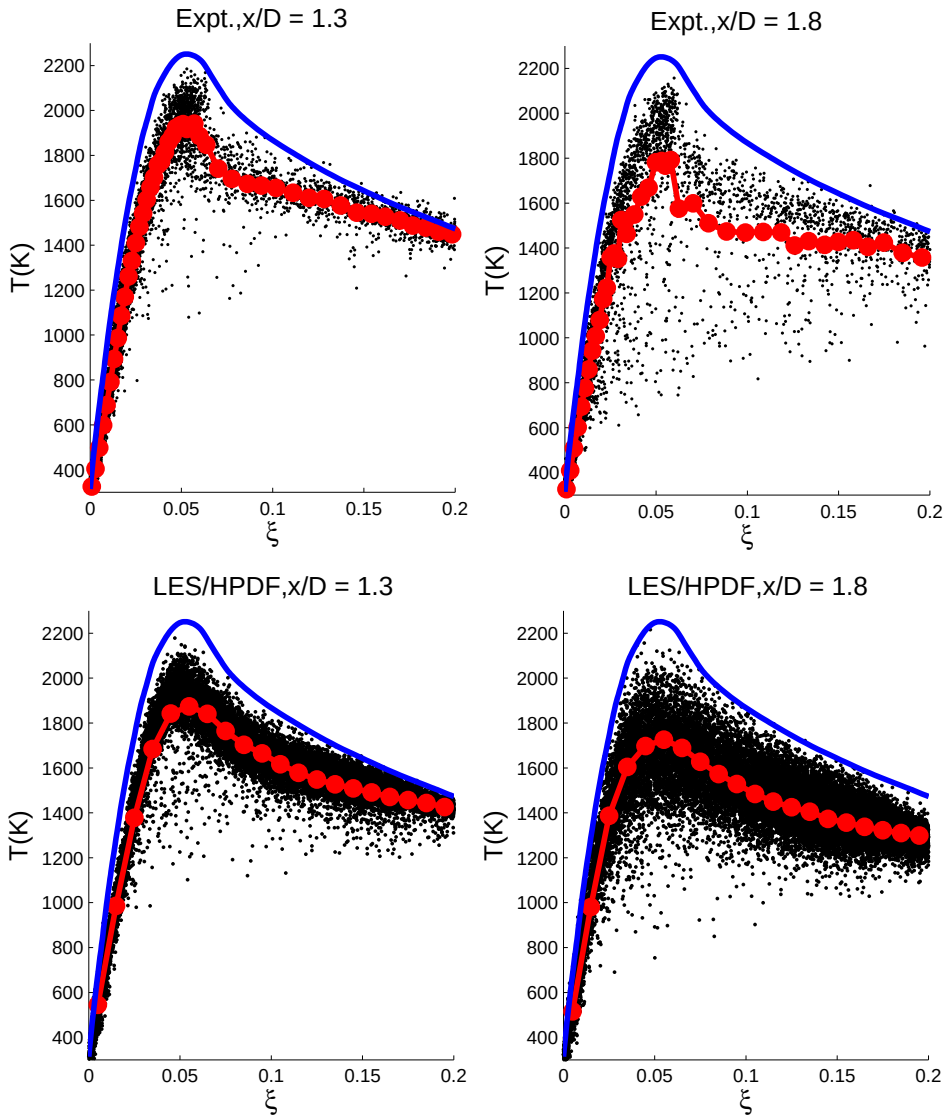

Figure 23: HM2 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 1.3D_B$ and $x = 1.8D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

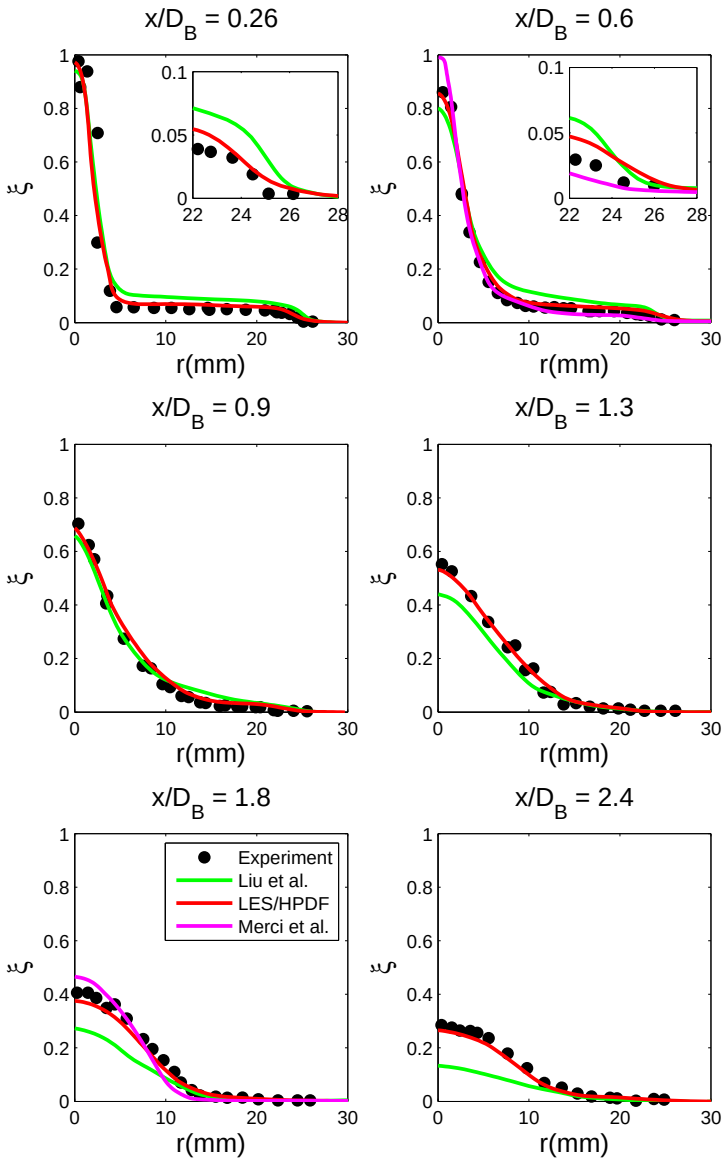

Figure 24: Radial plots of the Favre-averaged mixture fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

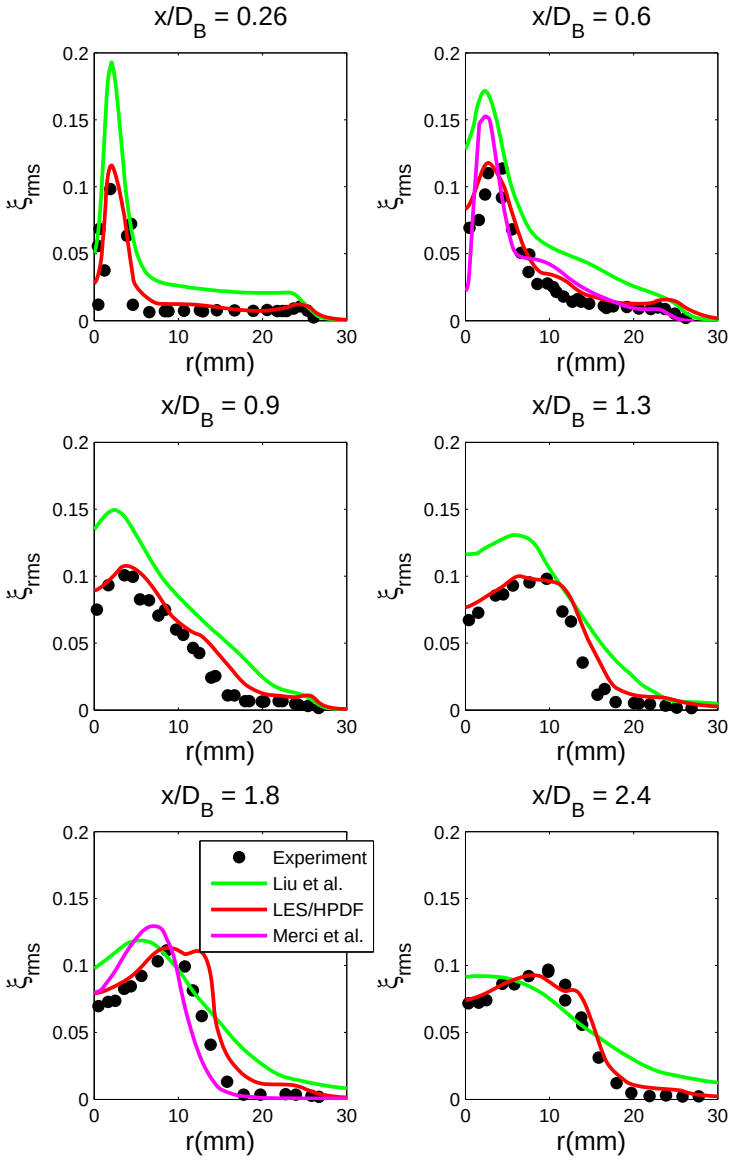

Figure 25: Radial plots of the root-mean-square of mixture fraction fluctuations at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

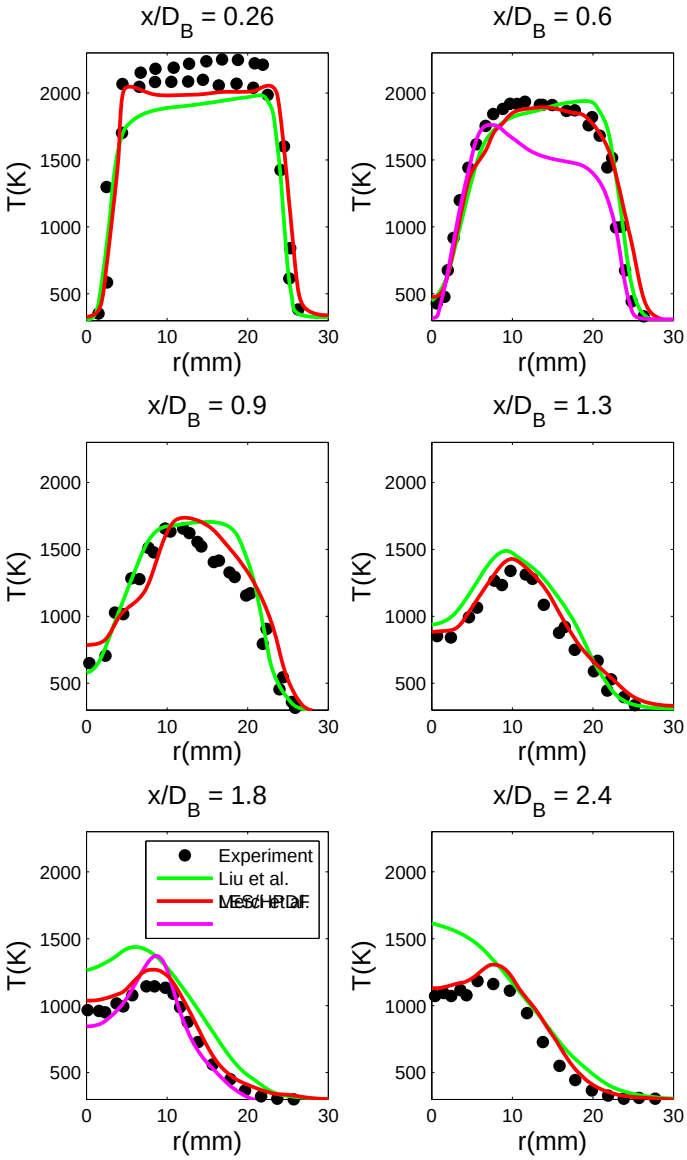

Figure 26: Radial plots of the Favre-averaged temperature at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

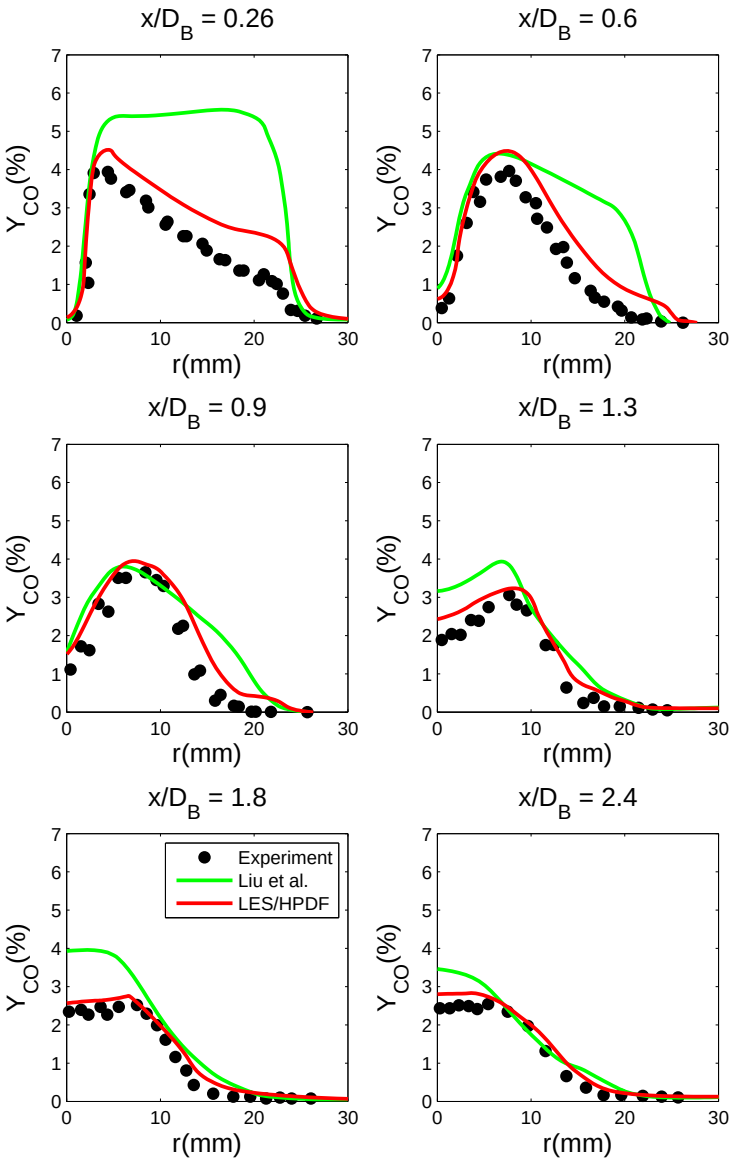

Figure 27: Radial plots of the Favre-averaged CO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

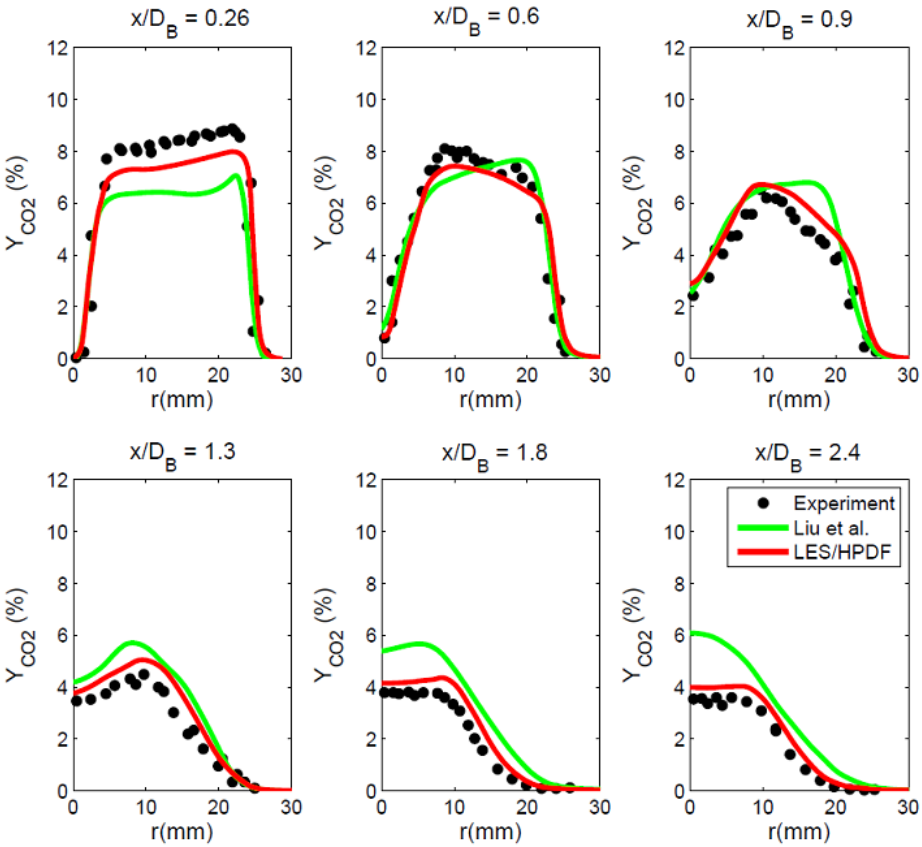

Figure 28: Radial plots of the Favre-averaged CO2 mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

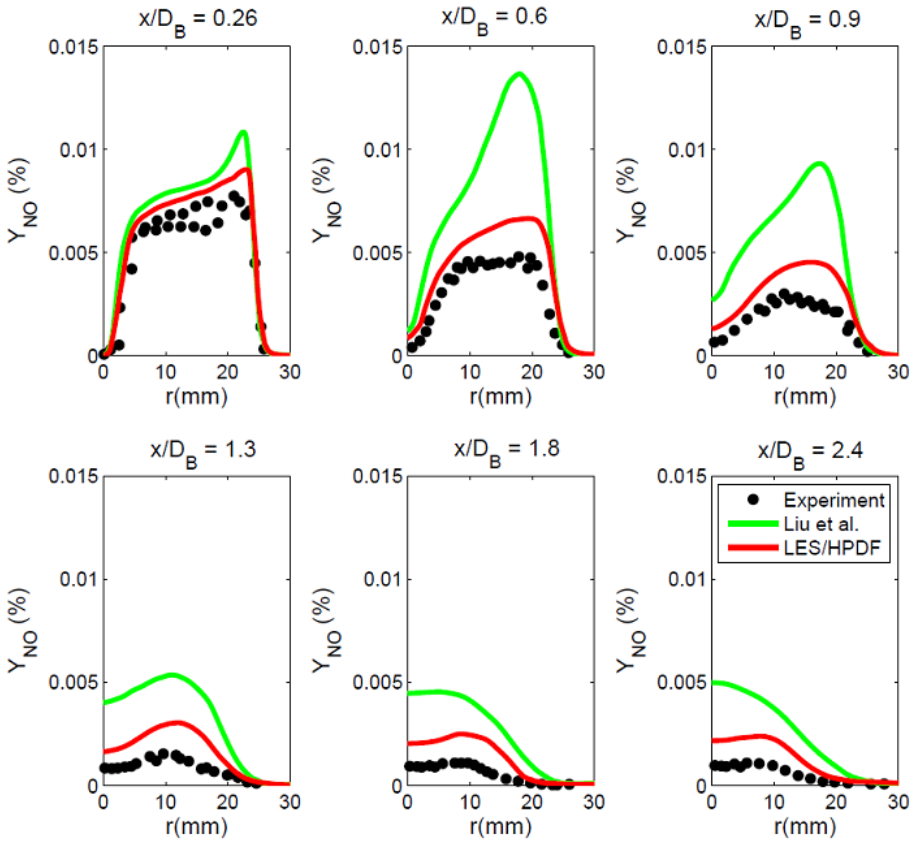

Figure 29: Radial plots of the Favre-averaged NO mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

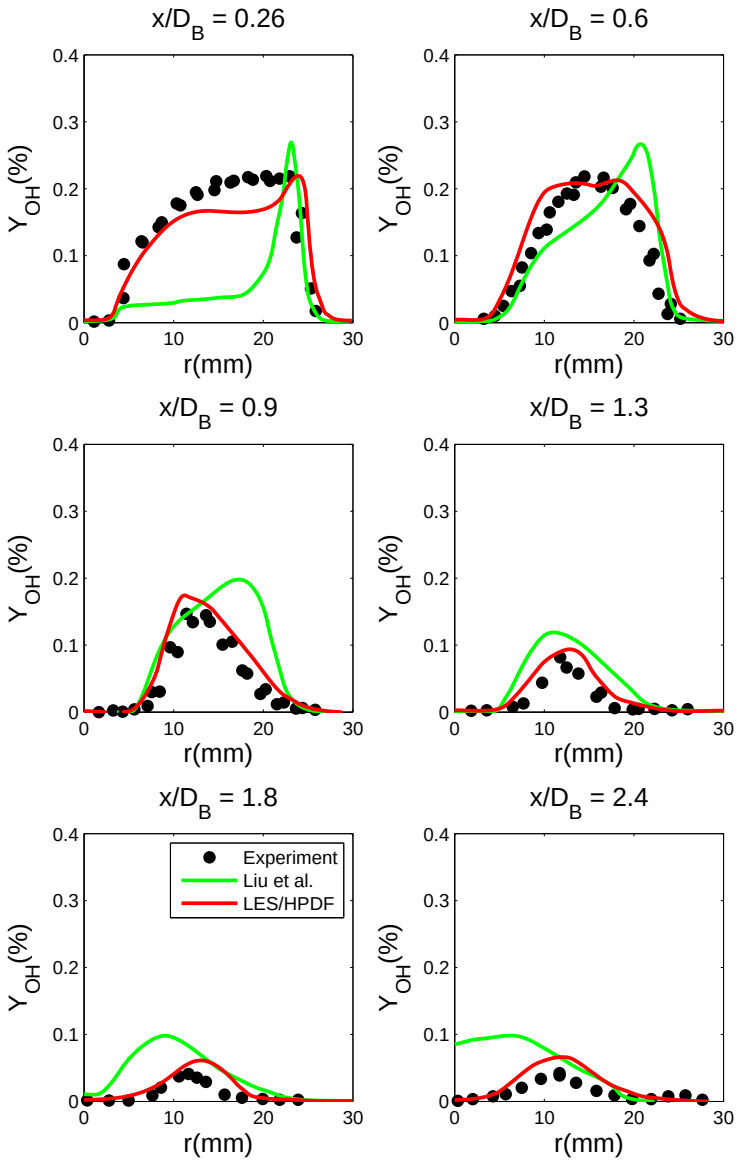

Figure 30: Radial plots of the Favre-averaged OH mass fraction at six axial locations, from $x = 0.26D_B$ to $x = 2.4D_B$, for the HM3 simulation.

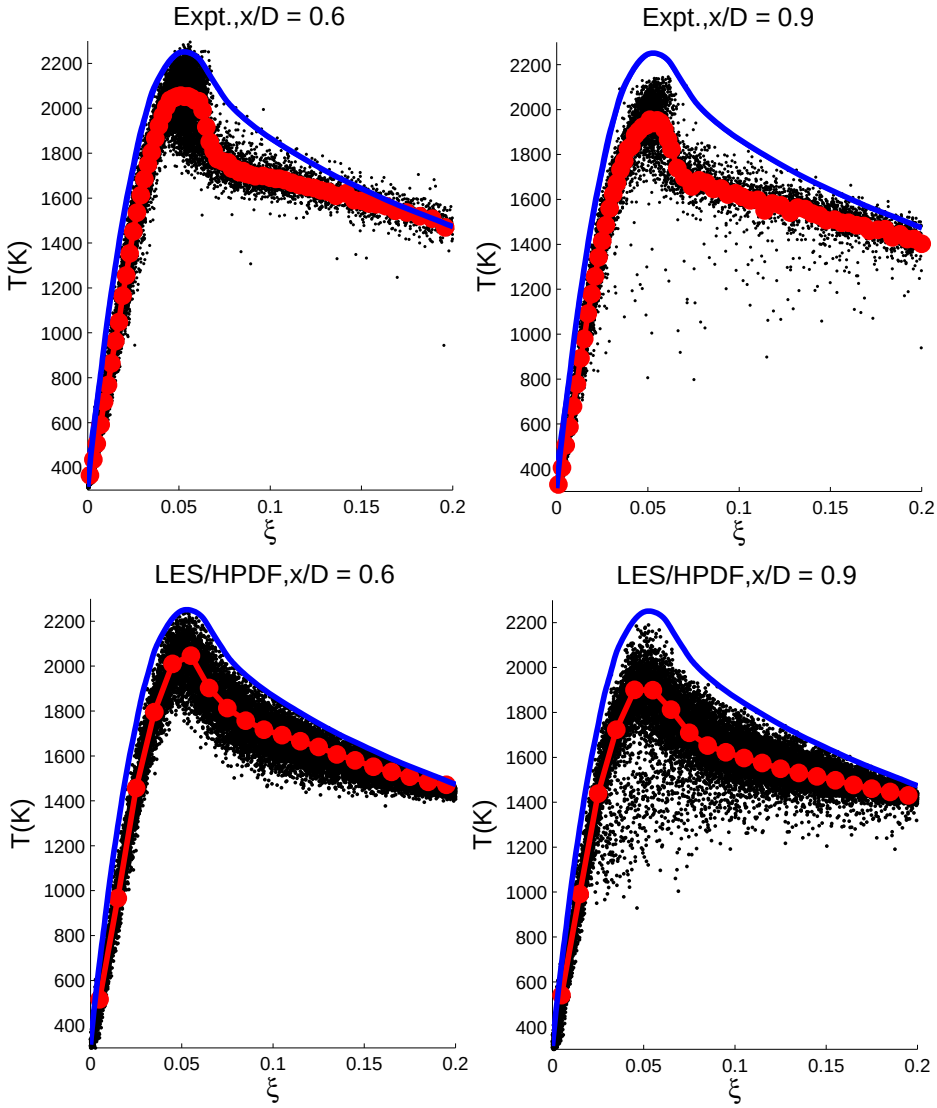

Figure 31: HM3 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 0.6D_B$ and $x = 0.9D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

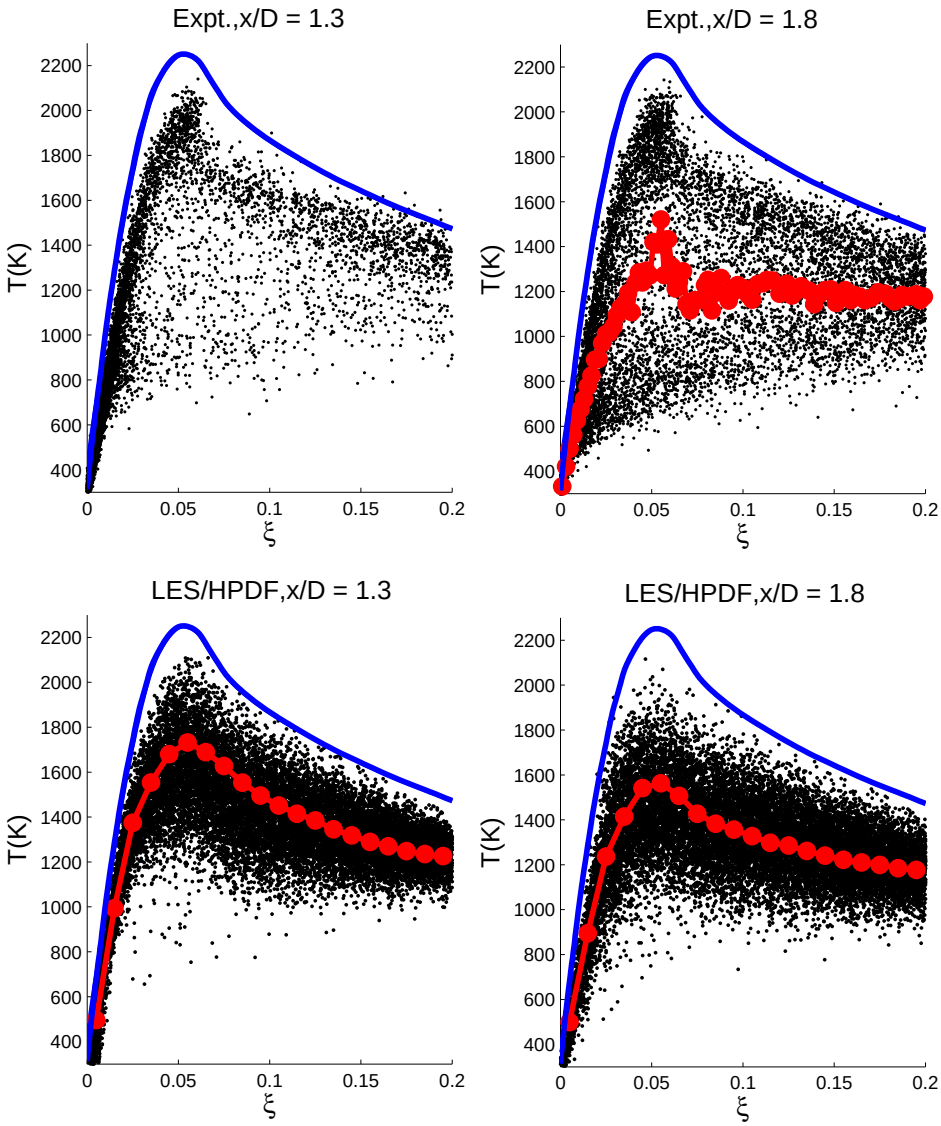

Figure 32: HM3 scatter plots of temperature plotted against mixture fraction, in the range $\xi \in [0, 0.2]$, for the axial locations $x = 1.3D_B$ and $x = 1.8D_B$. Conditional means are marked in red, and laminar flamelet temperatures are marked in blue. Top: experimental results. Bottom: LES/HPDF calculations.

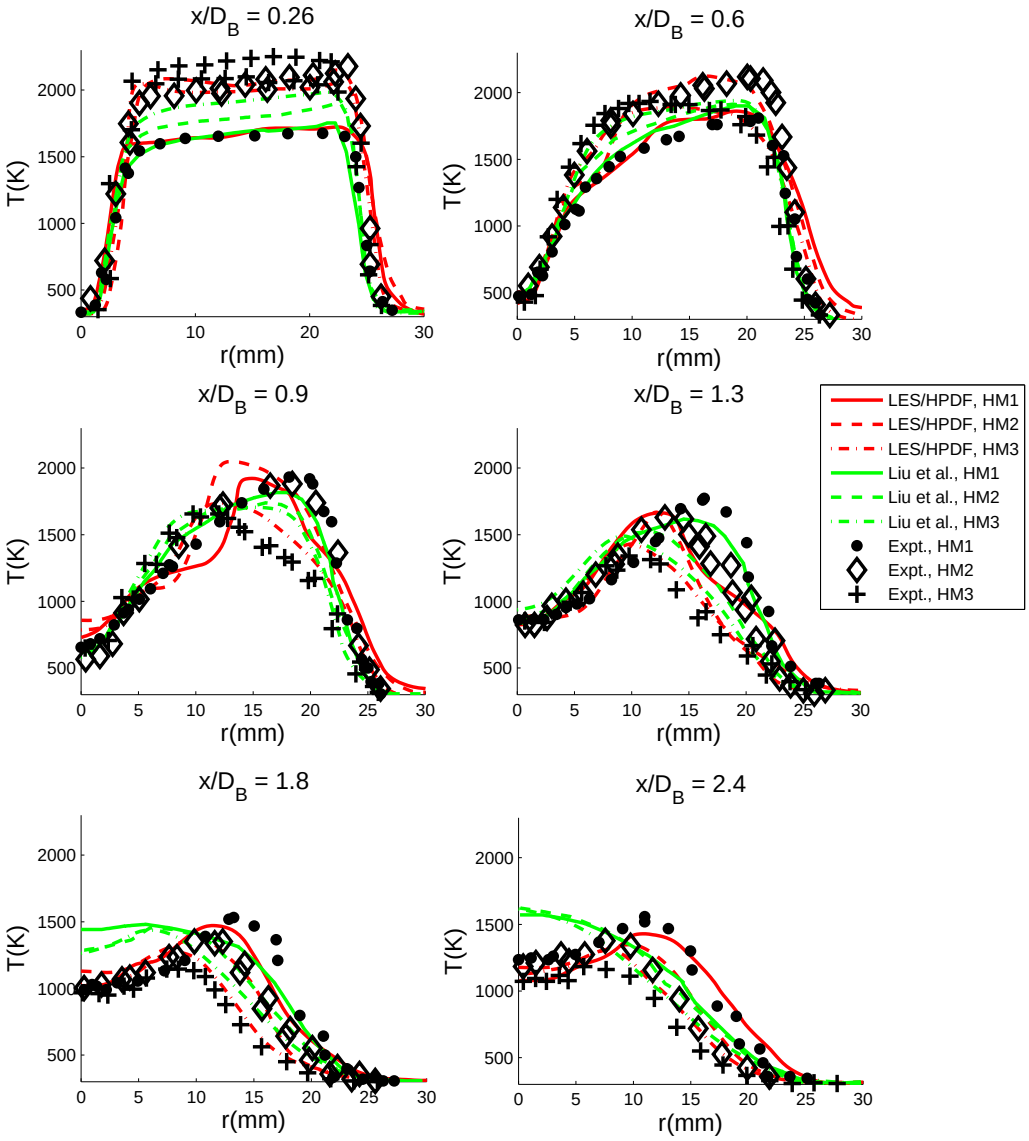

Figure 33: Comparison of the temperature profile variation from flame HM1 to HM2 to HM3. Symbols: experimental data. Red curves: LES/HPDF. Green curves: Liu et al.

Conclusions and a Look Ahead

This dissertation presents new advances in hybrid particle/finite volume methods for turbulent reactive flows. In the area of particle mass consistency preservation, new interpolation and time stepping schemes are introduced, and their effectiveness is shown to be better than that of existing numerical algorithms. A new time integration scheme, MRK2, is designed for the evolution of particle positions when advected via a discontinuous velocity field: MRK2 improves upon standard time integration schemes in that it preserves the continuity of the Lagrangian mapping between the initial and final particle positions, and yields second-order convergence in the case when the time step decreases to zero for a fixed grid.

A new class of SDE time integration schemes of arbitrarily high order of accuracy is proposed, and its convergence properties are proven rigorously for all members of the class, and demonstrated numerically for the second- and third-order cases. These schemes, called the DRp schemes, compare favorably, in terms of accuracy and computational efficiency, to standard second-order accurate SDE time integration schemes, such as those of Kloeden and Platen, and Cao and Pope.

In the context of the LES/HPDF code developed at Cornell and used for the simulation of the Sandia-Sydney Bluff-Body flames, specific volume coupling is examined in detail, and an overall second-order accurate LES/PDF procedure is achieved, in contrast to previous LES/PDF codes which are first-order accurate in time. The specific volume coupling error is minimized via a transported specific volume approach, which allows for an overall second-order accurate LES/PDF scheme while minimizing the statistical errors inherent in the PDF solver by using the LES component of the LES/HPDF code to account for the transport of specific volume.

Particle mass consistency errors need to be minimized in order for the LES/PDF procedure to yield a valid numerical approximation to the mass-weighted PDF evolution equation. This is done via a combination of second-order accurate time integration schemes used by the LES/PDF code, new velocity and scalar interpolation schemes designed to reduce the build-up of particle mass consistency errors, and a corrective velocity algorithm which reduces these errors once they occur.

The LES/HPDF code, with chemistry modeled by the ARM2 chemical mechanism, is used to simulate the Sandia-Sydney bluff-body flames, HM1, HM2, and HM3. The simulation is seen to be sensitive to the treatment of heat transfer to the bluff body face: better agreement of the temperature profiles with experiment is achieved with the implementation of a Dirichlet temperature boundary condition. Overall, the agreement of the LES/HPDF calculations with experiment is very good, with the exception of the upstream regions of the outer shear layer, where the LES/HPDF-calculated scalars converge to the coflow value more slowly than experimental data and other computational results - this may be corrected by a higher grid resolution in the outer shear layer.

In general, the LES/PDF results are considerable improvement over previ-

ous numerical studies of this series of flames. In particular, for flames HM2 and HM3, which feature considerable flow complexity and local extinction, the qualitative and quantitative agreement of the LES/HPDF simulations with experiment is better than that of previous numerical studies, which used the RANS/PDF approach, of the same flames. This suggests that the LES/HPDF code is ready to be tested on other hydrodynamically-complex flows with turbulence-chemistry interactions.

There are several possible extensions of this line of research. While the MRK2 scheme deals effectively with discontinuous velocity fields, it is also desirable to develop new SDE integration schemes that are designed to deal with noisy and probably discontinuous diffusivity fields - in the LES/PDF context, the turbulent diffusivity evaluated by standard models such as the dynamic Smagorinsky procedure is almost always very noisy, which in the present LES/PDF context, with turbulent diffusivity and no constraints on the number of cells that a particle can cross in the radial and circumferential directions, suggests that there is potential for improvement of the numerical accuracy by accounting for the change in diffusivity experienced by those particles which cross into a different LES cell. On the same rationale, numerical accuracy could be improved by accounting for the change in mean composition and scalar mixing frequency seen by particles which cross one (or several) cell interfaces in the period of a single time step, similarly to the way that MRK2 accounts for the change in velocity experiences by particles which cross a cell interface.

The ultimate goal in the LES/PDF field is the development of a code which allows for accurate numerical simulations of complex engineering applications, such as for example internal combustion engines and turbines. The transition to such applications from the laboratory-scale gas combustion problems, to which the LES/HPDF code is presently suited, requires a plethora of obstacles to be overcome. The increase in grid sizes necessary to handle most engineering applications causes a considerable increase in computational cost, thus requiring improved computational efficiency of the LES/HPDF code and its use in ever-faster supercomputers: this is also necessitated by the fact that the hydrocarbon fuels used in typical applications are much more complex than the simple methane-hydrogen-air deals with in this work, and thus require more detailed chemical mechanisms.

We also note that many physical processes present in turbulent combustion are currently not simulated by the LES/HPDF code. Modeling of these processes is also an important avenue for the extension of the present work. In particular, an accurate calculation of droplet vaporization is essential for the development of numerical algorithms for liquid fuel combustion. Additionally, radiation and conduction within adjacent solids may play a significant role in certain applications, and does in fact affect even the bluff-body flames simulated in this work: as noted above, the accuracy of the bluff-body temperature profiles is increased by a Dirichlet boundary condition for temperature. This condition, however, is not predictive, in that it requires the experimentally-measured temperature values on the bluff-body face - in order for these temperature values to be obtained numerically, treatment of the heat exchange at the bluff-body face,

including convection, radiation and conduction to the interior of the bluff-body, would be necessary.

Additionally, the irregular geometry of most applications implies the necessity to adapt the LES/HPDF code unstructured grids, as opposed to the Cartesian and cylindrical grids which the code uses at present. Note, however, that this may also be achieved via use of the immersed boundary method.

Perhaps the most immediate possibility for the extension of the present work lies in the inclusion of near-wall treatment for the LES/HPDF wall interfaces. In the present work, the lack such near-wall treatment is not a considerable obstacle since the boundary layers which develop in the inlets, and separate at the bluff-body corners, are simulated by a highly-resolved incompressible pipe flow solutions. However, in a future application, which may include solid structures within the LES/HPDF computational domain, with turbulent boundary layer development on their walls, the inclusion of a near-wall LES model for the accurate calculation of these boundary layers will be essential.

To summarize, there are many advances which need to materialize in the field of LES/PDF methods for turbulent combustion in order to get to the ultimate goal of an algorithm which accurately simulates complex engineering applications: the studies performed within this dissertation are but one of these advances.